ETD Archive

2009

Uncertainty Management of Intelligent Feature Selection in Wireless Sensor Networks

Sanchita Mal-sarkar
Cleveland State University

UNCERTAINTY MANAGEMENT OF INTELLIGENT FEATURE SELECTION

IN WIRELESS SENSOR NETWORKS

SANCHITA MAL-SARKAR

Bachelor of Science in Physics

Calcutta University, India

April, 1987

Master of Science in Physics

Benaras Hindu University, India

December, 1989

Master of Computer Science

University of Windsor, Canada

June, 1999

submitted in partial fulfillment of requirements for the degree

DOCTOR OF ENGINEERING

at the

CLEVELAND STATE UNIVERSITY

December, 2009

DEDICATION

To my beloved daughter,

Tatini

ACKNOWLEDGMENT

First and foremost, I express my sincere gratitude to Dr. Vijay Konangi and Dr. Iftikhar Sikder for introducing me to this field and being my advisors over the years. It would not have been possible to write this doctoral dissertation without their guidance, support, and patience all throughout. I would like to acknowledge my other committee members: Dr. Chansu Yu, Dr. Wenbing Zhao, and Dr. Santosh K. Misra for their critical reviews and fruitful discussions that improved the quality of this dissertation.

I extend my special appreciation to Dr. Mary Jane Saunders, Provost and Senior Vice President, for helping me at both professional and personal levels. Her generous help and support will be remembered forever. I also like to acknowledge the support and cooperation from my colleagues at Cleveland State University. Special thanks go to Dr. Barbara Benander, Dr. Alan Benander and Dr. Mary Murray McDonald for their help and encouragement.

I am grateful to Dr. Lotfi Zadeh, whom I first met at a conference in Akron. I was and continue to be astounded by his kindness and knowledge. He always responded immediately to my queries, either by telephone or by e-mail. His great help and advice are deeply appreciated. I would also like to acknowledge

This dissertation has been approved

for the Department of ELECTRICAL AND COMPUTER ENGINEERING

and the College of Graduate Studies by

_____

Dr. Vijay K. Konangi, Dissertation Committee Co-Chairperson

_____

Department/Date

_____

Dr. Iftikhar Sikder, Dissertation Committee Co-Chairperson

_____

Department/Date

_____

Dr. Chansu Yu

_____

Department/Date

_____

Dr. Wenbing Zhao

_____

Department/Date

_____

Dr. Santosh K. Misra

_____

Department/Date

UNCERTAINTY MANAGEMENT OF INTELLIGENT FEATURE SELECTION

IN WIRELESS SENSOR NETWORKS

SANCHITA MAL-SARKAR

ABSTRACT

Wireless sensor networks (WSN) are envisioned to revolutionize the paradigm of monitoring complex real-world systems at a very high resolution. However, the deployment of a large number of unattended sensor nodes in hostile environments, frequent changes of environment dynamics, and severe resource constraints pose uncertainties and limit the potential use of WSN in complex real-world applications. Although uncertainty management in Artificial Intelligence (AI) is well developed and well investigated, its implications in wireless sensor environments are inadequately addressed. This dissertation addresses uncertainty management issues of spatio-temporal patterns generated from sensor data. It provides a framework for characterizing spatio-temporal pattern in WSN. Using *rough set theory* and temporal reasoning a novel formalism has been developed to characterize and quantify the uncertainties in predicting spatio-temporal patterns from sensor data. This research also uncovers the trade-off among the uncertainty measures, which can be used to develop a multi-objective optimization model for real-time decision making in sensor data aggregation and sampling.

TABLE OF CONTENTS

x

LIST OF TABLES

LIST OF FIGURES

CHAPTER I
INTRODUCTION

1.1 Chapter Introduction

This chapter introduces the research problems and challenges in developing the uncertainty management scheme of wireless sensor networks (WSN). The research motivation of this thesis is derived from limited support for the uncertainty management of traditional approaches in WSN. Some motivating applications are described in the context of spatio-temporal pattern-based data aggregation and uncertainty management. The research questions are formulated from the statement of problems and the research scopes are outlined. The contributions of this thesis are summarized. The research methodology and the validation of its contributions are also briefly described.

1.2 An Overview of Wireless Sensor Networks

The popularity of an array of wireless devices, such as PDAs (Personal Digital Assistants), palmtops, cell phones, and laptops results from the motivation of being untethered and yet to be connected. There are two complementary technologies for connecting these devices: cellular data networks that connect these devices to the internet using base stations, and ad hoc networks that connect the devices through multi-hop wireless networks without using base stations. Wireless sensor networks is an ad hoc networks that consists of a large number of inexpensive, low-powered, multi-functional sensor nodes, each equipped with a sensing circuit, a digital signal processor, and radio links (Akyildiz, Su, Sankarasubramaniam, & Cayirci, 2002). Furthermore, each sensing circuit includes one or multiple sensing devices, such as acoustic, seismic, still or motion video camera, infrared (IR) or magnetic sensors.

Often, the design and the deployment of sensor networks are application specific, unlike traditional data networks such as the Internet. Broadcasts, or multicast operations, are fundamental to the realization of these networks. In a typical problem scenario, several thousands of sensor nodes are deployed, either manually or by robots, and each sensor node communicates with a few other neighbors. They communicate in an energy-efficient manner within radio communication range using radio links, in order to collectively establish an ad

hoc network. The sensor nodes operate and respond to a very dynamic and often harsh environment; they adapt their over-all sensing accuracy to resource-constrained (e.g. energy, bandwidth, and memory) environments.

Sensor networks are envisioned to revolutionize the paradigm of collecting and processing information, both in an urban environment as well as in inaccessible terrain. They can be used for monitoring complex real-world systems at a temporal and spatial granularity which was not previously possible (Bhaskar Krishnamachari, 2005). They are one of the most rapidly developing new information technologies that have the potential to be used for a wide range of applications: health monitoring, military surveillance and target tracking, ecological habitat monitoring, environmental monitoring, and industrial sensing and diagnostics.

However, the deployment of a large number of unattended sensor nodes in hostile environments, frequent changes of the environment dynamics, and severe resource constraints pose uncertainties that prevent WSN from meeting its full potential in real-life applications. Another challenge is its interdisciplinary nature – research in sensor networks requires contributions from signal processing, networking, database and information management, data mining, GIS (Geographic Information System), machine learning, AI (Artificial Intelligence), and distributed algorithms and architectures.

Sensor networks are expected to provide large scale, yet fine-grained coverage by employing a large number of inexpensive sensor nodes. However, these inexpensive nodes can be unreliable and error prone. Besides, nodes are often deployed in harsh environments such as, forest fires or underwater. It is important to ensure that the system performance will not degrade below a certain threshold, despite its individual node failures. The uncertainties in every aspect of the system need to be identified and quantified. These include sensed data uncertainty, actuator uncertainty, sensor status uncertainty, limited sensing ranges, compromised nodes, channel mal function, transmission collision, imprecision in localization and synchronization, topology and routing uncertainty, mobile uncertainty, and resource uncertainty, and so forth. (Y. Liu & Das, 2006). Visions of large numbers of nodes will remain unrealized in practice until some uncertainty handling mechanism is formally incorporated in model to ensure its robustness and reliability.

1.3 Research Problem Statement

Albert Einstein said, "The mere formulation of a problem is often far more essential than its solution." The definition and articulation of problems is a critical task for analysis and design of a system and should be performed in a systematic order within a system framework. It allows one to develop a complete and comprehensive understanding of the nature of a problem, underlying physical phenomena, and processes for achieving a set of objectives. The

boundary of a system is determined from the mission, goals, and objective of the analysis, and performance (Ayyub & Klir, 2006).

There is an increasing awareness that there are several phenomena and problems, such as uncertainty, that cannot be well explained within the boundaries of individual disciplines of science. Uncertainty is an important measure in the analysis of risk. In recent years, it has been recognized that uncertainty management is an interdisciplinary research area and should be formally considered in decision making at all levels in a system framework.

In real life, sensor data streams are coming from different sources at different space and times and they can be unbounded for constantly evolving entities, such as temperature, pressure, etc. The challenge is how to aggregate these unbounded data streams at different space and times and provide decisions making in real-time. Traditional data processing techniques are not suitable for streams data processing since unbounded data streams cannot be stored on the processing system on entirety. The inability to store an entire data stream suggests the use of some form of approximations. As a consequence of approximation in data aggregation, it is not always possible to obtain the exact or precise results and uncertainty may result.

Pattern discovery plays an important role in data aggregation. By identifying generic patterns from data streams without human supervision, pattern discovery algorithms can extract the most relevant information with high fidelity and remove the irrelevant patterns. Such pattern based data aggregation schemes have a potential to significantly reduce data communications when data has spatial and temporal correlations. However, the challenges of pattern discovery for target objects include that the data sets are, in general, not task-specific and the features collected for target objects are not always relevant for their classifications and hence should be pruned out or filtered. It is difficult to construct an information feature space because of the uncertainty about the relevance of the features. Several unsupervised clustering approaches are proposed in the literature, such as hierarchical clustering, k-mean clustering, self-organizing maps, and so forth. The goal of these approaches is to partition the datasets into statistically meaningful classes. Rough set theory is a well-known formalism for feature selection and rule generation. By employing rough set formalisms on dataset, whether numeric or symbolic, one can determine the features that are redundant and the features that are most relevant to a given application.

Besides there are several inherent issues in WSN such as, the deployment of a large number of unattended sensor nodes in hostile environments, frequent changes of environment dynamics, and severe resource constraints, that

contribute to the significance of uncertainties in WSN and prevent its potential to revolutionize many segments of life and economy. It is difficult to completely capture the nature of uncertainty in WSN and to cover all of its aspects because of its complex nature, application domain dependency and its propagation through all epistemological levels of a system by varying degree (Ayyub & Klir, 2006). In order to make wise decision even in the presence of uncertainty, it is imperative to characterize and quantify the potential uncertainties. To characterize uncertainties in WSN, the first step is to determine the sources of uncertainties for a particular application domain. Once the sources of uncertainties are identified, one may identify and quantify the uncertainties by employing existing formalisms, or by developing new formalisms.

The uncertainties in WSN stem from the missing or unreliable data. Missing data may arise during sensor reading, data sampling, format conversion, data discretization, data aggregation, feature selection, data routing, data savings to storage devices, incorrect data labeling, and so forth. Unreliable data may result from random noise, actuator uncertainty, sensor status uncertainty, limited sensing ranges, compromised nodes, improper channels, transmission collisions, routing uncertainty, or resource uncertainty (Kargupta, 2007; Y. Liu & Das, 2006).

The problem of characterizing uncertainty in sensor networks is that it requires in-depth knowledge of philosophical foundations of uncertainty, conceptual

frameworks underlying the uncertainty analysis, and technical methods to realize the uncertainty analysis (Ayyub, 2003). Philosophically, uncertainty can be classified into ontological uncertainty and epistemic uncertainty. Broadly, ontological uncertainty in sensor networks can be attributed to two distinct sources: the uncertainty regarding the frequency of distinct samples to be covered by the network, and the number of observable attributes to be monitored by each sensor node. On the other hand, epistemic uncertainty stems from a lack of complete knowledge and can be reduced at the cost of increased resources (I. U. Sikder, 2003).

Uncertainty in sampling frequency can be handled by a means of statistical estimation, such as simulation of space-time sample distribution, by using Monte Carlo simulation and domain specific *a priori* distribution. However, reducing the dimensionality of attributes is a difficult task because the dataset size needed to approximate a multivariate function grows exponentially with the number of variables of the function (Bellman, 1961). A high dimensional attribute-oriented sensor networks can adversely affect communication as well as data processing performance (e.g., training in learning systems).

Moreover, many real world systems exhibit non-polynomial complexity with respect to attribute dimensionality. For example, a large scale water treatment plant may require a huge number of attributes to be monitored through sensors

for performing diagnostic fault detection. The impact of removing even a few attributes can make a significant improvement in inference speed. The impact of removing 4 variables from NP-hard inference engines (given $\mathbf{O}$ (2^n)) results in the increase of the inference speed by a factor of 16. Moreover, the costs associated with maintaining connections to diagnostic computing equipments can be reduced since the points of failure (malfunctioning sensors or overly noisy sensors) are reduced significantly (Chouchoulas, 2001; Shen & Chouchoulas, 2000).

Sensor networks, capable of reducing multidimensional features at the node level, have immense potential of handling complex learning algorithms, producing significant patterns, and at the same time reducing energy consumption for data transmission. The shortcomings of the conventional hill-climbing approaches to feature selection are well documented (Jensen & Shen, 2004). In particular, traditional feature reduction approaches tend to change the underlying semantics of the sensor parameters after reduction, such as transformation-based approaches (Devijver & Kittler, 1982). For example, principal component analysis (PCA) (Jolliffe, 1986), a canonical means of data transformation and feature reduction, irreversibly destroys the underlying semantics of the dataset. It can only deal with linear projections, ignoring any non-linear structure in the data. Moreover, only purely numerical (non-symbolic) datasets may be processed by PCA. Further, reasoning about the data is almost

never possible, prohibiting the use of PCA as a pre-processor for symbolic or descriptive fuzzy modeling and other approaches dependent on data semantics. Hence, the application of PCA in sensor networks is limited.

1.4 Research Questions

The current research identifies the following questions based on the problem statements provided in the previous section:

- Can we provide a formalism for pattern-based data aggregation techniques in WSN that has the potential to reduce data communications as well as to support rule mining for symbolic, quantitative, and outlier data?
- Can we establish a formalism that would incorporate uncertainty, stemming from imprecision and vagueness of attributes, in the context of spatio-temporal patterns discovered from sensor data?
- Can we develop an algorithm to generate spatio-temporal patterns and reason about the "part and whole," or spatio-temporal *mereological* relationship, of the network while preserving the semantics of the sensor attributes?
- Can we characterize and qualify data uncertainty from the spatio-temporal patterns generated from the time series of sensor data in real time?

- Can we uncover the correlations among the uncertainty measures and determine whether the correlation is statistically significant in the context of WSN?

- In collaborative processing, how can we measure the dependencies of sensor attributes and identify redundant as well as indispensable parameters considering the space-time dependency and data aggregation?

The research questions identified above are not necessarily mutually exclusive. These questions have been narrowed down within the limited scope of data aggregation issues in wireless sensor networks.

1.5 Research Motivation: Some Application Scenarios

The current research can be applied to a number of domains:

- forest fires

- aquatic biodiversity mapping

- precision agriculture

The first scenario is based on the environmental problem of determining the most critical climatologic conditions; for instance, temperature, relative humidity, and wind speed that contribute to the forest fires, and find their spatiotemporal patterns. The second motivating example includes the problem of

determining the quality of water or biodiversity by measuring its characteristics, such as temperature, density, salinity, acidity, chemicals, conductivity, P_H and oxygen. The current research can determine the redundant attributes and find the spatiotemporal patterns for aquatic biodiversity. The third motivating scenario is based on the problem of finding the spatiotemporal patterns in precision agriculture. In all the applications mentioned above, it is also possible to indentify and quantify different types of uncertainties from the spatiotemporal patterns so that they can be incorporated in the model parameters. With a model optimized in the context of uncertainty, one can make more realistic decisions.

1.5.1 Early detection of Forest Fires

Forest fires often start unnoticed and spread very quickly, causing millions of dollars in damage and claiming many human lives every year in the United States as well as Canada. For example, on June 21, 2008, a lightning-sparked forest fire started in the scenic town of Big Sur in Northern California and quickly turned into an uncontrollable firestorm due to a strong wind. More than 1,200 fire fighters, several thousand armed forces, and several hundred national guards and volunteers were unable to stop the firestorm even after two weeks of battling. It burned more than 1,100 square miles and destroyed about 100 homes. Officials have said this unprecedented fire season, plagued by drought and high temperatures, has seen the most fires burning at any one time in recorded California history.

The National Weather Service predicted more dry lightning toward the end of the week of the California disaster; however, forecasters did not expect such a severe firestorm, when nearly 8,000 lightning strikes sparked about 800 fires. Early detection of hot spots and appropriate measures could prevent, or, at least minimize the damage and casualties. Common causes of forest fires are lightning, extreme hot and arid weather, severe drought, and human unawareness. The lack of a high resolution forest fire detection system prevents forecasters from predicting this disaster early on, with high accuracy.

Current forest fire detection systems are based on satellite imagery; they take a longer amount of time, typically one to two days, to provide a complete image of the Earth. Additionally, satellite-based systems cannot detect fire unless it spreads at least 0.1 hectare and its location accuracy is 1 km (Li, Nadon, & Cihlar, 2000) (Lohi, Ikola, Rauste, & Kelha, 1999) Another limitation of the satellite-based system is that its efficiency is affected by weather conditions, such as clouds, fogs and rains. Although it can scan a large area at one time, it cannot detect forest fires with high accuracy in real time because of its low resolution and long period of scan.

With the advancements of wireless sensor networks, it is anticipated that real time forest fire detection systems can be developed using wireless sensors data, with high precision and accuracy (Hefeeda, 2008). Thousands of disposable

sensors can be densely scattered over a disaster prone area to form a wireless sensor network in a forest. The sensors collect environmental data, such as temperature, relative humidity, soil moisture, barometric pressure, wind speed, and wind direction - potential attributes to forest fires and its spread. Sensed data is reported to the processing center, and the processing center, after analyzing the data, directs the appropriate agency to dispatch fire fighters and/or to evacuate local residents, depending on the extracted information from sensor data.

Sensor nodes are self-organized into geographic clusters and one node in a cluster is designated as cluster head by employing a distributed algorithm. Since a cluster head requires more communications than other sensor nodes in the cluster, the role of the cluster head is rotated among the sensor nodes to prolong network lifetime. Communication within a cluster is either single-hop or multi-hop. Data, before sending to the data processing center, can be aggregated locally at each cluster head to reduce communication, to save energy, power and bandwidth, and to improve data quality. By analyzing the data stream, aggregated at cluster level, homogeneous patterns can be generated and sent to the sink for decision making.

We propose to employ the notion of temporal templates on sensor data stream, gathered at each cluster head, to generate homogeneous spatio-temporal patterns

and the corresponding *if-then* rules from temporal information systems for determining potential hot spots in near real time. It is possible by generating temporal templates from the change of attributes, and comparing them with the temporal templates generated from the historical data. For example, when the change of temperature suddenly gets very high and the humidity is very low, the chance of forest fires is very high. To the best of our knowledge, this is the first attempt to use the notion of temporal templates in wireless sensor networks. The satellite data is useful to determine the spatiotemporal patterns; however, the sensor based temporal template can generate the patterns with high precision and accuracy.

Analyzing spatio-temporal patterns of data at each time step is not possible because of the huge time series. It is important to find the area of the data where the changes to the spatial patterns are most likely to occur and to focus the analysis in this portion of the data to find the spatiotemporal patterns preceding the forest fires. Before generating spatio-temporal patterns, we can reduce the number of attributes, and determine the most important attributes for forest fires, by employing the indiscernibity concept of rough set theory.

Our model, equipped with the uncertainty handling mechanism, works even when data is imprecise, incomplete, or ambiguous, which is common in resource constraint sensor networks, consisting of inexpensive, unreliable and error prone

nodes. Besides, often times nodes are deployed in harsh environments (e.g. forest fires) and some of the sensor nodes may fall into the fire zone and be destroyed, or they may die because of energy depletion and are unable to send data. Sensor nodes can also send ambiguous data because of low power levels. Sensors within the burn zone, before being scorched, can record the change of attributes, such as, the increase of temperature or the decrease of barometric pressure and humidity as the flame front advances. It is important to ensure that the system performance will not degrade below a certain threshold, despite its individual node failures.

1.5.2 Aquatic Biodiversity Mapping

The application scenario is an aquatic sensor network-based, biodiversity mapping. Underwater sensors can be used to determine the quality of water, or biodiversity, by measuring its characteristics, such as temperature, density, salinity, acidity, chemicals, conductivity, P_H, oxygen, dissolved methane gas and turbidity (Akyildiz, 2005). A sensor can sense single or multiple attributes and multiple sensors can sense the same attribute. The sensors in a cluster are equipped with domain specific function procedures or lookup tables with limited computing capability. A sensor node that can reach the maximum number of sensor nodes in one hop is selected as a cluster head; it broadcasts an advertisement to all other nodes in the network. However, there are several

algorithms that can be used to determine the cluster head, depending on the

residual energy of the node, maximum number of neighbors, and so forth.

The cluster head nodes gather data from all ordinary nodes or non-cluster-head

nodes, aggregate data, and send the data to the sink or base station. Cluster-head

nodes consume more energy than non-cluster-head nodes because the cluster-

head needs to receive data from all cluster members in its cluster and then send

the data to the sink. The cluster heads are selected in each round to make sure

that the energy consumption is evenly distributed among all the sensor nodes to

prolong network lifetime. The scheme uses TDMA MAC for intra-cluster

communications and CDMA for inter-cluster transmissions. TDMA has two

phases: the setup phase to organize the clusters and the steady-state phase to

allow all nodes to periodically transmit during their time slots.
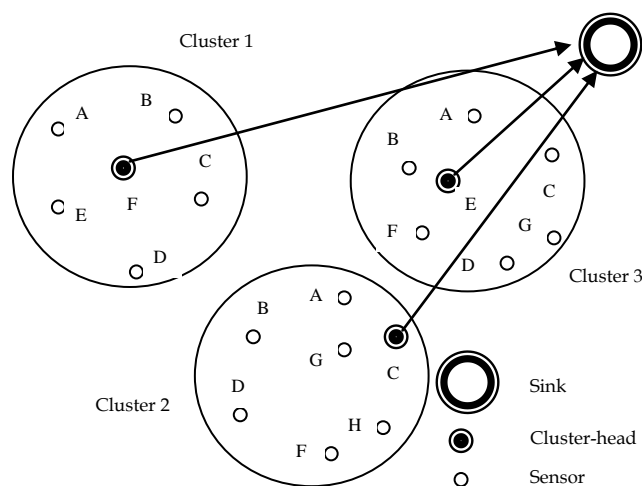
Figure 1.1: A series of geographic clusters

18

Since the data is processed locally and only the result is sent to the sink, this data aggregation technique decreases energy consumption during data transmission. Figure 1.1 represents a series of geographical clusters. F is selected as the cluster-head in cluster 1 because it has the highest number of nearest neighbors (A, B, C, D and E). Similarly, node C and node E are selected as cluster-heads for cluster 2 and cluster 3 respectively. Using rough set formalism, we show that it is possible to reason over uncertain data, stemming from incomplete and inconsistent information received from spatially distributed sensors.

1.5.3 Precision Agriculture

Precision agriculture, an agricultural concept relying on the existence of in-field variability, is one of the most promising application areas of wireless sensor networks. It can be useful for irrigation management, frost detection and early warning, pesticide application, harvest timing, and water quality measurement and control ("Crossbow Technology Inc.," 2005). Researchers are working to gain a better understanding about the potential of sensor networks in precision agriculture. It has been shown that the variability within a site is substantial and often cannot be predicted by statistical models ("Intel Research," 2005; "Intel Research," 2008).

Precision agriculture, instrumented with wireless sensor networks, is capable of providing detailed and site-specific knowledge on the crop production variability and management by exploiting their spatiotemporal variability of soil, light, temperature, and nutrition stress. It promises higher yield and lower production costs by streamlining (standardizing) and centralizing agricultural management, hence reducing the labor cost ("NanoTechnology," 2007). It can help feed the expanding population of the world by increasing the viability of semi-arable lands ("Intel Research," 2005).

Advances in wireless sensor technology have made the practical deployment of various site-specific services possible, which until recently were considered extremely costly and labor intensive. By employing wireless sensors, it is possible to monitor the plants in real time on a much finer level than traditional precision agriculture techniques. Real time, yet fined-grained information of the field will provide a solid base for farmers to remotely monitor agricultural practices, and to adjust strategies accordingly at any time. The real time climatologic data and environmental data, such as air temperature, relative humidity, soil moisture, light, wind speed, wind direction, leaf wetness and nutrition stress are sensed by heterogeneous sensor nodes and relayed to a cluster node. Then the cluster node can perform pattern-based data aggregation and send the patterns or rules to the central repository for decision making.

By employing a real time decision support system (DSS), we can make critical decisions on when, where and how to apply water, fertilizer, lime, and pesticides. For example, Phytophtora is a common fungal disease in potato fields. The development and the spread of this disease strongly depend on the climatologic conditions, such as humidity and temperature. Another important attribute that helps develop fungal infection in a crop field is leaf wetness (Baggio, 2005). By recording these critical conditions with the use of sensors, it is possible to determine which part of the crop field is at high risk for developing fungal disease and where to apply fungicide. They can also detect frost and send early alerts on potential damage of the crop.

Another potential application includes a vineyard that consists of different regions with different types of soil, and different levels of soil moisture, sun exposure, and temperature. The regions with heavy clay soil and less exposed to sun (in a shed) can hold water for a longer time, while the regions with sandy soil that are more exposed to the sun are likely to dry out faster. It is important to monitor soil moisture at different depths at each location since water moves through the soil at different rates ("NanoTechnology," 2007) . With precision knowledge of the soil moisture status at different locations and different depths, it is possible to minimize water use while optimizing the yield and quality of grapes, which is critical for dry seasons.

The success depends not only on the ability to collect data by employing sensor networks, but also on the ability to analyze the data even in the presence of data uncertainty and to apply it for real time decision making in agricultural practices.

1.6 Research Scope

The sensors are assumed to be static and have some processing capabilities. The sensor nodes may not be homogeneous in terms of power or memory. For example, the nodes designated as cluster head may be more powerful compared to other regular sensor nodes. In order to generate a useful number of spatio-temporal templates, attribute values should not change very frequently and they must have some spatial and temporal correlations.

1.7 Research Contribution

The current research is expected to produce significant contributions which may have practical implications to deal with the intelligent feature selection, data stream processing, knowledge discovery, and uncertainty management in wireless sensor networks. In particular, the specific contributions are as follows:

- Formalization of a novel pattern-based data aggregation technique that has the potential to reduce data communication, due to the fact that only the rules are sent to the sink.

- o Discovery of spatio-temporal patterns and RS rule induction from sensor data stream by intelligent feature selection.

- o Support for symbolic, quantitative, and outlier data mining.

- Foundation of uncertainty management in real time sensor database systems for constantly-evolving entities.

- Characterization and quantification of uncertainties in wireless sensor networks environments by introducing a hybrid model of uncertainty management based on rough set theory and Dempster Shafer theory of evidence.

- Demonstration of trade off among the uncertainty measures in the context of WSN.

1.8 Validation of Research Contribution

The research proposed in the thesis is based on a formal approach derived from the approximate reasoning methods of rough set theory (Pawlak, 1992b) in the context of the uncertainty management of sensor networks. The proposed method is applied on NOAA's TAO/TRITON sensor data, and spatio-temporal patterns are generated from the time series of the dataset. Since we are not aware of any publicly available real data generated by WSN, effort is made to simulate sensor network data by performing some preprocessing steps, such as spatial clustering, from the time series of TAO/TRITON sensor data in order to increase data correlation and data homogeneity. In the context of current research, data correlation and data homogeneity are the two important aspects of WSN that are

exploited in discovering spatio-temporal patterns and rule generation. Moreover, our research is not only restricted to WSN, it is applicable for any environments that involve constantly evolving sources of data and data correlation. In general, most research in the area of WSN use simulated data since it is difficult to obtain real sensor data.

We use rough set feature selection technique to remove the irrelevant patterns that have no impact on decision making. This step is important to generate a fewer number of compact rules from the data, given that the sensor data set has spatio-temporal correlation. The use of rough set theory for the feature selection is well recognized in literature (Swiniarski & Skowron, 2003). Then the rule validation is performed on different parts of the patterns taken in random sequence to reduce the bias. The rule validation ensures that the rules faithfully represent the data set and the rules are self-consistent. This is important in the context of current research because only the rules are periodically sent to the sink where the critical decision is made only based on the rules. It is to be noted that the proposed model is a pattern-based data aggregation model, not a prediction model.

The uncertainties in spatio-temporal patterns stemming from missing and inconsistent data are characterized and quantified by logical extension of rough set theory in the WSN environment. Some well-established uncertainty

measures, such as entropy and nonspecificity are redefined in the research context and some new uncertainty measures, such as time window, pattern quality, pattern variance, pattern frequency variance, and so forth are established in the context of pattern generation from constantly evolving entities. Also, the sensitivity analysis is performed on the uncertainty measures and their statistical significance is demonstrated.

1.9 Research Methodology

The research reported in this dissertation identifies a research problem that is not only important for wireless sensor networks, but also for many real systems because uncertainty is an inherent attribute of real systems. Some applications described allow for current research to be very useful.

As the first step, the existing approaches in the context of the current research are critically examined, and their limitations in the context of current research are outlined. Then, the formalism for the current research is developed by enhancing and integrating the existing theories. The methodology also identifies the boundary conditions in the context of the current research and provides the research results by employing the formalism on sensor data within the research boundary. As a result, a series of spatio-temporal patterns are discovered from the datasets. Uncertainties in the patterns are characterized and quantified by employing the formalism on the patterns. The methodology also provides the

scheme on rule validation for spatio-temporal patterns and sensitivity analysis performed on uncertainty measures. The statistical validation of the correlation of the uncertainty measures is established. Finally, it provides the summary of the results, conclusion, and possible future work in the direction of current research.

1.10 Organization of the Dissertation

The remainder of the dissertation is organized as follows: Chapter II explains related literature and provides a critical review in the context of pattern generation and uncertainty management in WSN. Chapter III provides the methodological foundation of the current research in terms of mathematical formalism development and rule validation. Chapter IV provides validation of the mathematical formalism for rule generation and uncertainty management, as well as the significance of the research findings. Chapter V concludes with a brief summary of the current research, research significance, and potential future direction of the research.

CHAPTER II
LITERATURE REVIEW

2.1 Chapter Introduction

This chapter examines the context and relevance of the existing approaches to pattern-based data stream aggregation and uncertainty management in Wireless sensor networks. In this chapter, we provide an extensive literature review of data stream processing, data aggregation protocols, soft computing approaches of pattern generation and rule induction, and uncertainty handling formalisms, in the context of wireless sensor networks and uncertainty management. Each section concludes with a brief summary of the literature review for that section, along with an explanation of the direction and the motivation of the current research. Finally, the chapter ends with a critical summary of the literature review in the context of uncertainty management of WSN.

2.2 Data Stream Processing in WSN

A data stream is a real-time, continuous, and ordered (implicitly by arrival time or explicitly by timestamp) sequence of elements. In many applications - such as network monitoring, financial analysis, clickstream monitoring, manufacturing, wireless networks, radio frequency identification (RFID), and sensor networks - data takes the form of continuous data streams rather than finite stored data sets and clients require long-running continuous queries instead of one-time queries. The traditional data processing techniques may not work because of the unbounded nature of data stream. The differences between traditional data mining and stream data mining are as shown in Table 2.1 (Kargupta, 2007; Ulrych, 2008).

Table 2. 1: Traditional vs. Stream Data

	Traditional Data Processing	Stream Data Processing
Storage capacity	Unbounded	Bounded
Number of passes	Multiple	One or very few
Type of result	Accurate	Approximate
Update type	Arbitrary modifications	Append
Distributed	No	Yes
Query type	One time, transient	Continuous, persistent
Data Representation	Relations	Time series

A sensor network can be modeled as a distributed system of data streams, a sequence of data elements which arrive online. It is not possible to control the order in which the elements arrive, nor is it possible to store the sensor data stream on the processing system in its entirety since the number of data elements

in a stream can be unbounded and larger than the storage capacity of the stream processing system. Thus, the query processing systems need to process these elements as soon as they arrive without storing and making multiple passes over it.

2.2.1 Traditional and Sensor Network Stream Data Processing

Data stream processing systems in WSN need to address new issues and challenges, which may not be required in traditional stream processing systems. The major differences between the traditional data stream processing and the data stream processing in WSN are as follows (Elnahrawy, 2003; Kargupta, 2007):

- Spatial and temporal attributes are important in sensor networks applications, whereas they have no major impact on data stream in other applications, such as web log and click streams. In wireless sensor networks, it is important to know the exact time and exact location for an application, such as forest fire.

- Data duplication in WSN is common since a large number of sensors are deployed densely in the environment monitor the similar environment. This correlation in data should be accounted while developing data mining techniques.

- Sensor data streams are often more noisy compared to traditional data streams because of resource constraints, the harsh environment of sensor

deployment, and the large number of inexpensive sensors. Data cleaning and uncertainty management are important to ensure the reliability of WSN.

- In the WSN environment, another major issue is energy efficiency, whereas in traditional stream data stream processing energy is not a major concern.

- In WSN, in-network aggregation is the acceptable mode of operation because of its severe resource constraints. On the other side, traditional data stream processing does not need to deal with these constraints and is performed outside of the network.

- Distributed processing of queries is the computational model in WSN, whereas centralized data stream processing serves as a basic model in traditional data stream management systems.

Learning in the dynamic environments with large number of distributed sources (sensors) of continuously evolving data, in addition to severe resource constraints of sensors, requires intelligent analysis of the data sets describing real world problems like weather forecasting or web log processing. Processing data streams requires adaptive algorithms that can deal with concept drift and are capable of incorporating new information and disregarding outdated information since the data collected from the sensor field may shift from time to time after some minimum permanence.

Algorithms, systems, and frameworks that address all these issues and challenges have been developed in recent years from the existing statistical and computational approaches. The approaches can be categorized into two groups, such as task-based and data-based. In task-based approaches, the techniques are adopted from computational theory so that they are space and time efficient. On the other hand, the data-based approach relies on the selection of a subset of the whole dataset or vertical or horizontal transformation of the dataset to an approximate and smaller data representation (Gaber, Zaslavsky, & Krishnaswamy, 2005).

2.2.2 Task-based Approach of Data Stream Processing: Time Window

Task-based approaches involve modifying existing techniques or developing new ones that address all computational challenges and issues of stream data processing in WSN mentioned in the previous section. There are several task-based approaches, such as approximate algorithms, sliding windows, and algorithm output granularity (Gaber, Zaslavsky et al., 2005). Approximate algorithms provide an approximate solution with error bounds for computationally hard problems. However, it is not appropriate for resource hungry WSN unless supported by other tools (Cormode & Muthukrishnan, 2005). The algorithm output granularity (AOG) is the resource-aware data processing approach that supports high and fluctuating data rates (Gaber, Krishnaswamy, & Zaslavsky, 2005). The motivation of the sliding window

approach is the applications where people are interested in the most recent data streams, instead of old streams. We will mainly focus on this sliding window in the context of the current research, where most recent patterns are of utmost importance.

Sliding Window

Because of the unbounded nature of the sensor data stream, we cannot store an entire data set on the stream processing systems. Thus, the important issue is which part of the data stream is selected for stream mining. There are three data stream processing models, such as absolute (or fixed), landmark, and sliding windows. The absolute model mines all frequent itemsets over the entire history of stream data within a specific time window where both the start and the end of the window are specified.

The Landmark model mines all frequent itemsets from a specific time to the present. This model is not appropriate when the most recent information of a data stream is of only importance, such as stock monitoring systems. The sliding window model mines all frequent itemsets contained in the sliding window and both ends of the window change with time as it slides. The data items within the sliding window are considered fresh. The information is discarded when the timestamp expires and they are out of the range of the sliding window. In general, the size of the sliding window depends on the applications and system

resources (Jiang & Gruenwald, 2006). There is also another model called damped model which mines frequent itemsets in stream data by assigning more weight to most recent transactions compared to older ones (Chang & Lee, 2003).

Most of the research in the data stream model is based on the assumption that all elements in the data stream are equally important and representative; however, in sensor networks applications, this assumption is not always true. In many sensor network applications, recent data elements are more useful than old data elements, and we can use sliding window model, where each element of the data streams expires after N (window size) time steps and the set of last N elements is the relevant part of the data stream for gathering statistics, generating rules or answering queries (Elmagarmid, Ghanem, Hammad, Mokbel, & Aref, 2007).

2.2.3 Data-based Approach of Data Stream Processing

Data-based techniques involve either the summarization of the whole dataset or the selection of a subset of the whole data stream. The most commonly used methods for data reduction are data sampling, synopsis and histograms, principal component analysis (PCA), and wavelets. Sampling, an old statistical technique, involves probabilistically selecting some tuples while others are skipped for data processing. We can sample instances at some periodic intervals, and, thus, if the rate of streams arrival exceeds the capacity of processing systems, sampling can manage this situation. It can be used to find approximate

results when the most expensive operation in database processing, namely join, is required (Kargupta, 2007). It executes random samples and then joins over samples instead of entire streams. However, the traditional sampling algorithm is not appropriate for stream data processing because traditional sampling requires the total number of tuples to be known in advance. Another disadvantage of sampling is that it does not support the application where data rates fluctuate (Gaber, Zaslavsky et al., 2005). A large number of sampling algorithms have been developed recently for stream processing by enhancing the traditional sampling algorithms mostly with the time based sliding window mechanism (Gibbons & Tirthapura, 2002). Reservoir sampling is a well-known technique for sequential random sampling over data streams (Vitter, 1985). But the conventional reservoir sampling assumes a fixed size reservoir, and it does not work with moving windows.

Synopsies and histograms are summarization techniques that can be used to compute the frequency distribution of stream elements, and they can be used in range queries (Babcock, Babu, Datar, Motwani, & Widom, 2002). A histogram is defined by a set of non-overlapping intervals, defined by the boundaries and frequency counts. The sketch, a special case of synopsis, vertically samples the incoming data stream and does not maintain the sequence of sensor streams. Hence, augmented with hashing and time-based windowing schemes, sketch support stream processing.

Another data reduction technique is PCA, a mathematical tool for transforming a number of possibly correlated variables into a smaller number of uncorrelated variables, called principal components (PC). The first component represents the variability as much as possible in data and the subsequent components represent the variability as much as possible in the remaining data. (Kargupta, Puttagunta, Klein, & Sarkar, 2006; Y. Zhang & Edgar, 2006) suggest the use of PCA as a data reduction technique in sensor data stream processing. However, PCA can process only numeric data (non-symbolic data), destroys the underlying data semantics, and only can deal with linear data.

Another data reduction tool is wavelets. Wavelets transforms are used in several applications since most signals can be represented by small set of coefficients (Chakrabarti, Garofalakis, Rastogi, & Shim, 2001; Guha & Harb, 2005). The limitations of wavelets are as follows: they do not work for symbolic data and they only work for numerical data. Besides the technical disadvantage, the Haar wavelet is not continuous and thus not differentiable. Though the research in wavelets is a fast growing area, there are some unresolved issues. "For example, what is the best choice of wavelet to use for a particular problem? How far does the harmonic wavelet transform computational simplicity compensate its slow decay rate in the x-domain? How it can be used for the solution of integrodifferential equations, and many others? The disadvantage of harmonic

wavelets is that its decay rate is relatively low (proportional to x^{-1}), therefore, its localization is not precise" (Cattani & Kudreyko, 2008).

2.2.4 Rough Sets and Feature-Selection-based Data Reduction

Rough set theory is a well-known formalism for feature selection and rule generation. By employing rough set formalisms on datasets, whether numeric or symbolic, one can determine the features that are redundant and the features that are most relevant to a given application. Thus one can determine the set of sensors that participate in decision making. This is very important for stream data processing in WSN, where space, energy, bandwidth, and time are crucial.

2.2.5 Synopsis of Existing Data Stream Processing Techniques

The inability to store an entire data stream suggests the use of some form of approximations, such as synopses. As a consequence of approximation, it is not always possible to obtain the exact or precise results and uncertainty may result. Thus in order to make wise decisions, even in the presence of imprecision, it is imperative to identify potential uncertainties that stem from these approximations of sensor data streams in a particular application domain. Once we can identify the type of uncertainties, we can quantify them either by employing the existing mathematical formalisms or by developing new formalisms that are appropriate for stream data processing. However, the

literature review shows that the existing strategies barely address uncertainty as one of the important dimensions in the optimization space for data stream processing. Given the growing demands for complex domain specific applications of sensor networks such as object tracking and event identification, high resource constraints of sensor nodes and the characteristics of sensor data streams, it is important to deal with the uncertainty and the uncertainty propagation through stream data processing in a real time environment. Therefore, the existing stream processing algorithms for wireless sensor networks should be enhanced by incorporating uncertainty handling mechanisms in their model parameters.

There are several issues that need to be addressed while mining stream data in WSN. First, a compact data structure is needed to store, retrieve, and update the information because of the bounded memory requirement. There are several data structures - such as tree, array, and link lists - that can be considered depending on the applications. The second issue is the granularity of approximate results. Third is the speed of processing. The fourth concern is uncertainty or error. The fifth one is whether the data stream mining algorithm can handle a large amount of data. The sixth concern can be whether there are tradeoffs among them. For instance, high granularity in approximation may result in more resource consumption or processing delays.

One of the possible solutions can be to develop a formalism for sensor network data stream processing which exploits spatial and temporal correlation in sensor data stream and removes irrelevant information by intelligent feature selection so that it satisfies the space, energy, bandwidth requirements as well as incorporates uncertainty handling mechanisms. The notion of temporal templates (Synak, 2001), homogeneous patterns occurring at regular intervals, can be used for processing sensor data streams. Spatio-temporal patterns can be extracted from temporal information system by employing the concept of sliding windows and then, *if … then …* rules, can be generated using a formalism, rough set theory, which is data driven and inherently equipped with uncertainty handling mechanism formalism. It is expected that there is a strong regularity in sensor data for some intervals since the data is gathered continuously from densely populated sensors.

2.3 Data Aggregation in WSN

Data aggregation plays an important role in severely resource constraint WSN, where a large number of sensors are densely deployed in an unfriendly environment. In wireless sensor networks, data is gathered at intermediate points from multiple sources and periodically transmitted to the sink or base station for processing. The amount of data gathered at the sink could be overwhelming if all sources send the data directly to the sink. Besides this, there

is a high probability of redundancy and correlation in data since data is often collected from densely deployed neighboring sensors. Not only sensor data have spatial correlation, but also they have temporal correlation since sensors are continuously monitoring the environment. By exploiting these correlations in data while aggregating data at intermediate sensor nodes, a significant amount of savings in terms of data communication, energy, and bandwidth can be achieved. Other advantages of data aggregation at intermediate nodes include improvement in data quality, reduction in overall traffic in the network and network delay, improvement in the performance, and reduction in the power consumption in transmitting information. In-network data aggregation is considered an effective technique in sensor networks since the communication cost is often much higher compared to the computation cost (Kargupta, 2007).

Data aggregation is application-dependent, and the data aggregator depends on the target application. Sensor network applications have different needs, different traffic patterns (one-to-many, many-to-one and many-to-many), and different data rates (fixed and variable, frequent and infrequent). There is no algorithm that matches the requirements of all applications. Energy savings depend on the type of aggregator.

The important issue of data aggregation is where to conduct the processing of the data. If data is processed locally and only the result is sent to the sink, it will

increase the energy consumption by the processor and decrease the energy consumption during data transmission. If raw data is sent to the sink for processing, it will decrease the energy consumption in processing but increase the energy consumption of data transmission. The problem of determining the optimal selection and location of aggregation points in sensor networks is NP-complete, in general. However, attempts have been made to find an approximate solution for the data aggregation problem.

The design of efficient data aggregation techniques is an inherently challenging task. Network life time, data accuracy, data freshness and latency are some of the important measures of data aggregation schemes. Timing plays an important role in determining data accuracy and data freshness; the important decision is how long a node should wait to receive data from its downstream nodes before forwarding to the sink or base station. Longer waiting time increases data accuracy but decreases data freshness. We can save a significant amount of energy by proper selection of data aggregation and forwarding intervals. There is a trade-off among network life time, data accuracy, data freshness, and latency (Solis & Obraczka, 2003).

2.3.1 Existing Data Aggregation Methods in WSN

Data aggregation techniques can be broadly classified into two categories, such as data-centric and hierarchical. Unlike traditional networks, it is not feasible to

assign global identifiers to each node of the sensor networks since the number of nodes is often very large and thus it is hard to select a set of sensors to be queried. Data is generally transmitted from every sensor node within the deployment region with significant redundancy, which is very energy inefficient. This leads to the development of data-centric routing, which is capable of selecting a set of sensor nodes and performing data aggregation as well as data routing without using nodes' addresses. The advantages of data centric routing include energy savings through data negotiation and the elimination of redundant data. The major drawback of the data centric routing is that it can result in excessive communication and computation at the sink node. Besides that, sink failure can bring the entire network down. The concept of a hierarchical approach was developed to enhance scalability and energy efficiency of the sensor networks. The advantages of hierarchical data aggregation methods include routing simplicity, lower latency, and implementation of node heterogeneity. However, the routing is not always optimal with a hierarchical approach, where as optimal routing is guaranteed in data-centric scheme at the cost of overhead (Rajagopalan & Varshney, 2006).

2.3.2 Data Centric Algorithms

In data centric routing, the base station sends queries to certain regions and waits for data from sensors located in those selected regions. Two important dada-centric protocols are Sensor Protocols for Information via Negotiation (SPIN) and

Directed Diffusion. SPIN is the first data-centric protocol which is a push diffusion protocol for data collection and dissemination (W. Heinzelman, Kulik, & Balakrishnan, 1999). Later, Directed Diffusion was developed and became very popular (Intanagonwiwat, Govindan, & Estrin, 2000). Then, several algorithms have been developed either based on Directed Diffusion (Braginsky & Estrin, 2002; Chu, Haussecker, & Zhao, 2002; Schurgers & Srivastava, 2001) or similar ideas (Manjeshwar & Agrawal, 2001; Sadagopan, Krishnamachari, & Helmy, 2003; Shah & Rabaey, 2002; Yao & Gehrke, 2002).

SPIN

In SPIN, the sources initiate the diffusion and the sink responds to the sources. The initiating node that has new data advertises the data to the neighboring nodes in the network using the metadata, description of data. A neighboring node interested in that data sends a request to the initiator node for data. The initiator node responds and sends data to the sinks. It only sends the requested data and avoids the cost of sending data needlessly; however it incurs the overhead associated with the negotiation phase. It uses point-to-point communication among pairs of nodes to eventually convey data to all interested participants. SPIN does not use an explicit aggregation mechanism and aggregation is performed implicitly during the initial negotiation between two nodes using meta-data to decide whether actual data will be exchanged (Solis & Obraczka, 2003).

SPIN incurs at a factor of 3.5 less energy consumption compared to flooding and is able to distribute 60% more data per unit of energy compared to flooding. SPIN is suitable for environments with mobile sensors since the forwarding decisions are based on local neighborhood information. One of the main advantages of SPIN is that topological changes are localized since each node requires the knowledge of its single hop neighbors. The main disadvantage of SPIN is its inability to guarantee data delivery. For instance, in intrusion detection applications, if the nodes interested in the data are further away from the source node, and the intermediate nodes are not interested in the data, then the data is not delivered to the nodes (Rajagopalan & Varshney, 2006).

Direct diffusion

In direct diffusion, a two-phase pull diffusion scheme, the sink broadcasts the query throughout the network with a rich set of attribute-value pairs describing the desired data to ensure that every node gets it (Intanagonwiwat et al., 2000). These queries are cached by the sensor nodes and data are sent to the sink once they are available to the sources (sensors). Gradients, reply links, are established between neighboring nodes during the broadcasting and used by the sources to select a path for data transmission to the sink.   There are several paths as exploratory data is transmitted to the sink from different gradients. The sink selects one of the multiple paths by sending message reinforcement and the real

data is transmitted using the selected path. Since the interest message, exploratory data, and reinforcement message are generated periodically, direct diffusion can accommodate node failures by updating the paths between the sink and the source.

Direct diffusion is appropriate for many sources and few sinks. The performance of data aggregation depends on the position of source and sink and on network topology. By checking the sequence of interests it receives, a node can find out its nearest neighbors. Even if the source and the sink are close to each other, many unnecessary communications are involved resulting in energy-inefficiency. If there is one source and multiple sinks, direct diffusion is not a suitable choice because it involves many unnecessary communications. Another drawback of direct diffusion is that it does not support applications that require continuous data transmissions to the sink. (B. Krishnamachari, Estrin, & Wicker, 2002) have shown the impact of source-destination placement and network density on the energy in data aggregation.

COUGAR

COUGAR is a data-centric protocol that uses declarative queries in order to abstract query processing from network layer functions, such as selection of leader nodes to perform in-network data aggregation and transmit the data to the sink. This abstraction is performed through a new layer between the network

and the application layers. The drawbacks of COUGAR include extra overhead due to additional query layers on each node, additional node synchronization, dynamic maintenance of leader nodes (Akkaya & Younis, 2005; Yao & Gehrke, 2002).

2.3.3 Hierarchical Algorithms

Hierarchical networks are comprised of two-layer routing where one layer is used to select cluster heads and the other layer is used for routing. The main idea of a hierarchical algorithm is to maintain the energy consumption of the sensor nodes by means of multi-hop communication within a particular cluster and performing data aggregation and fusion before the data sent to the sink (Akkaya & Younis, 2005). Examples of hierarchical algorithms include cluster-based algorithms and tree-based algorithms. LEACH is one of the first hierarchical routing algorithms for sensor networks where cluster formation is mainly based on the energy reserve of the sensors and the sensors' proximity to the cluster head (W. Heinzelman, Chandrakasan, & Balakrishnan, 2000). Later, several hierarchical algorithms have been proposed based on LEACH (Lindsey & Raghavendra, 2002; Lindsey, Raghavendra, & Sivalingam, 2002; Manjeshwar & Agrawal, 2001, 2002).

TEEN & APTEEN

Energy Efficient sensor Network protocol (TEEN) is based on a hierarchical approach, combined with a data-centric scheme useful for reacting in time-critical events. In the cluster formation phase, the closer nodes form clusters, and this process of forming clusters continues in several hierarchies until the sink is reached. After the cluster formation, the cluster head broadcasts two thresholds to the nodes, such as hard and soft thresholds. A hard threshold reflects the minimum possible value of an attribute for which the nodes are supposed to switch on their transmitter and send the information to the cluster head. On the other hand, a soft threshold provides the minimum possible change in the attribute values for which the sensor nodes are expected to transmit data to the sink (Akkaya & Younis, 2005). Thus, this protocol can control numbers of packets for time-critical applications. However, TEEN is not appropriate for the applications where periodic reports are required. Adaptive Threshold sensitive Energy Efficient sensor Network protocol (APTEEN) is an enhanced version of TEEN and APTEEN is useful for periodic data collections and reacting to time-critical events (Manjeshwar & Agrawal, 2001, 2002).

LEACH

W. Heinzelman et al (2000) propose a hierarchical clustering algorithm, Low Energy Adaptive Clustering Hierarchy (LEACH), where the cluster-heads

aggregate data and communicate directly with the sink or base station. LEACH

uses TDMA MAC for intra-cluster communications and CDMA for inter-cluster

transmissions. TDMA has been used to avoid collisions within a cluster and

CDMA supports simultaneous communications of cluster-heads with the sink.

The cluster-head node consumes more energy than the non-cluster-head node

because the cluster-head needs to receive data from all cluster members in its

cluster and then send the data to the sink. A node elects itself to become a

cluster-head by some probability and broadcasts an advertisement to all other

nodes in the network. A non-cluster-head node selects a cluster head based on

the signal strength. TDMA has two phases: a setup phase to organize the clusters

and a steady-state phase to allow all nodes to transmit periodically during their

time slots. The nodes send their data to the cluster-head and the cluster-heads

send the aggregated data to the sink at the end of each cycle. The cluster heads

are randomly selected in each round to make sure that the energy consumption

is evenly distributed among all the sensor nodes. There are several enhance

versions of LEACH, such as LEACH-C (W. R. Heinzelman, Chandrakasan, &

Balakrishnan, 2002) and LEACH-F (W. Heinzelman, 2000). In LEACH-C, the sink

broadcasts the cluster-head assignment to ensure even distribution of cluster-

heads in the sensor networks and to prolong network lifetime, whereas in

LEACH-F clusters are formed once and then they are fixed to reduce set-up

overhead at the beginning of each round.

PEGASIS

Power-Efficient Gathering in Sensor Information System (PEGASIS) is an enhanced version of LEACH (Lindsey & Raghavendra, 2002). Instead of forming multiple clusters, it forms chains from all sensor nodes so that each node transmits to and receives from its neighbors and the nodes take a turn as a cluster-head. The chain can be formed from a greedy distribution algorithm, or can be computed by the sink and sent to all the sensor nodes by broadcasting. It supports multi-hop routing by forming the chains and selecting only one node as cluster head that transmits to the sink (Akkaya & Younis, 2005).

EADAT

Energy Aware Distributed Aggregation Tree (EADAT) is proposed as an energy aware distributed heuristic to construct and maintain an aggregation tree in sensor networks (Ding, Cheng, & Xue, 2003). It is initiated by the sink which broadcasts the control message which includes five fields: sensor ID, parent, its residual power, status (leaf or non-leaf), and hopcount (number of hops from the sink). Data aggregation is performed only by the non-leaf nodes and the aggregated data is sent to the root node. The main idea is to turn off the radio of all leaf nodes to save power, and thereby extend the network lifetime. Therefore, in order to save the number of broadcasting messages, only the nonleaf nodes in the tree are responsible for data aggregation and traffic relay. EADAT algorithm makes no assumption of local network topology and is based on residual power.

It makes use of neighboring broadcast scheduling and distributed competition among neighbors (Rajagopalan & Varshney, 2006).

ESPDA

Cam, Özdemir, Nair, Muthuavinashiappan, & Sanli (2006) propose an energy-efficient secure pattern code based data aggregation scheme called ESPDA. It can prevent the redundant data transmission from sensor nodes to cluster-heads by implementing a pattern code based data aggregation. All except one of the sensor nodes are put into sleep mode, and pattern codes are generated from the sensed data. Cluster-heads compare patterns and send only distinct patterns, in encrypted form, to the sinks. Thus cluster-head do not need to know the sensed data for data aggregation. This process does not require the exchange of encryption and decryption key between sensor nodes and cluster-heads. This approach makes ESPDA energy efficient as well as secure. The authors in (Cam et al., 2006) have employed symmetric key cryptographic algorithm since WSN has limited energy, power, and storage.

Trust based Secure Data Aggregation

W. Zhang, Das, & Liu (2006) propose a trust-based framework, rooted in statistics and other distinct yet closely coupled techniques. Trustworthiness of individual nodes is evaluated by means of an information theoretical measure, Kulback-Leibler distance, and the compromised nodes are identified using an

unsupervised learning algorithm. Then an opinion, based on the degree of belief, is computed which represents the uncertainty stems from data aggregation.

2.3.4 Synopsis of Existing Data Aggregation Techniques

Many data aggregation techniques ignore the spatio-temporal correlation in sensor data. A significant savings in terms of data communication as well as processing can be made by exploiting such correlation (Kargupta, 2007). An overview of data aggregation techniques in wireless sensor networks is shown in Table 2.2 (Akkaya & Younis, 2005; Akyildiz et al., 2002; Rajagopalan & Varshney, 2006).

Table 2. 2: An Overview of Data Aggregation Tecniques

Authors	Methods (Categories)	Descriptions
W. Heinzelman et al.(1999)	SPIN (Data-centric)	Sends data to sensor nodes only if they are interested. Data aggregation is performed implicitly during the initial negotiation between two nodes using meta-data to decide whether actual data will be exchanged.
Intanagonwiwat et al.(2000)	Directed Diffusion (Data-centric)	Diffuses data through sensor nodes by using a naming scheme of the data. It sets up gradients for data to flow from source to sink during interest dissemination.
Yao & Gehrke (2002)	COUGAR (Data-centric)	Uses declarative queries in order to abstract query processing from network layer functions, such as selection of leader nodes to perform data aggregation and transmit the data to the sink.
Manjeshwar & Agrawal (2001; Manjeshwar & Agrawal, 2002)	TEEN & APTEEN (Data-centric & hierarchical)	Forms clusters in hierarchies from the closer nodes and this process continues in several hierarchies until the sink is reached. By means of soft and hard thresholds, TEEN can respond to time-critical events, whereas APTEEN is useful for both periodic data

		collections and reacting to time-critical events.
W. Heinzelman et al.(2000)	LEACH (Hierarchical clustering)	Forms clusters to minimize energy dissipation. It has two phases: a setup phase to organize the clusters and a steady-state phase to allow all nodes to transmit periodically during their time slots. The nodes send their data to the cluster-heads and the cluster-heads send the aggregated data to the sink at the end of each cycle.
Lindsey & Raghavendra (2002)	PEGASIS (Hierarchical clustering)	Forms a chain from all sensor nodes so that each node transmits to and receives from its neighbors and the nodes take a turn as a cluster-head. It is an enhanced version of LEACH and achieves significant energy savings compared to LEACH.
Ding et al.(2003)	EADAT (Hierarchical tree)	Forms a tree where data aggregation and data relay are performed only at non-leaf nodes and the aggregated data is sent to the root node designated as a sink. The algorithm is initiated by the root node which broadcasts a control message to inform about the sensor ID, its parents, its status, and its residual power, and hopcnt.
Cam et al.(2006)	ESPDA (Secure pattern code based)	Presents a secure energy-efficient data aggregation scheme that prevents the redundant data transmission from sensor nodes to cluster-heads by implementing a pattern code based data aggregation.
W. Zhang et al.(2006)	Trust based Secure Data Aggregation (Secure Trust based)	Proposes a trust based framework, rooted in statistics and other distinct yet closely coupled techniques. Trustworthiness of individual nodes is evaluated by means of an information theoretic measure, Kulback-Leibler distance, and the compromised nodes are identified using an unsupervised learning algorithm.

Regardless of the techniques employed, the existing strategies hardly mention one important dimension in the optimization space for data routing and data aggregation, namely uncertainty. Most of the data aggregation methods are optimized base on energy, power, or bandwidth. Recently some researchers addressed security and developed pattern-based model for data aggregation in

WSN (Cam 2006; Zhang 2006). However, they do not address uncertainty in a broader or more general frame. Given the growing demands for complex domain specific applications of sensor networks, unfriendly environment of sensor deployment, and severe resource constraints of sensor nodes, it is imperative to deal with the uncertainty and the uncertainty propagation through data aggregation in a real time environment. Therefore, aggregation-driven routing protocols for wireless sensor networks cannot optimize over only energy, power, bandwidth, or network life time – uncertainty should be included in optimization space. Thus a pattern-based data aggregation where energy, bandwidth, memory, security, uncertainty, and spatio-temporal correlation of data – are all addressed may be a potential solution for WSN.

2.4 Soft Computing Approaches for Rule Extraction

There are several soft computing approaches for rule mining, such as rough set theory, decision tree, and neural networks. Each of them has a distinct methodology for addressing problems in different application domains. We need to understand the issues better and determine or develop an efficient data mining algorithm in the context of streams and distributed systems.

2.4.1 Rough Set Theory and Rule Extraction

Rough set theory is very useful to discover hidden patterns in data and it is well recognized as a data mining tool (S. Mitra, Pal, & Mitra, 2002). It induces a set of

rules in IF-THEN form from decision tables. In rough set theory, data is represented in a two dimensional table or matrix called information tables (sometimes called a decision table). An information system (S) is expressed as $S = (U, A)$, where U is a nonempty finite set called the universe and A is a nonempty finite set of attributes. Each attribute a $\in A$ can be considered as a function that maps elements of U into a set V_a , where V_a represents the value set of attributes, such that

$$a: U \rightarrow V_a$$

For every subset of attributes $B \subseteq A$, one can associate equivalence or an indiscernibility relation I_B on U such that $I_B = \{(x, y) \in U: \forall a \in B, a(x) = a(y)\}$ and $I_B = \bigcap_{a \in B} I_a$. The B-lower and B-upper approximations are defined, respectively as:

$$\underline{B}X = \{x \in U: [x]_B \subseteq X\} \quad \text{and} \quad \overline{B}X = \{x \in U: [x]_B \cap X \neq \emptyset\}, \quad \text{where} \quad [x]_B$$

represents the equivalence class of the object $x \in U$ with respect to I_B.

X is B-exact or B-definable in S if $\underline{B}X = \overline{B}X$. The boundary region is represented as follows:

$BND_B(X) = \overline{B}X - \underline{B}X$. If $\underline{B}X = \overline{B}X$, the boundary region is empty. The boundary region is the set of objects that we cannot determine with certainty whether a member or nonmember of X. The rules induced from the boundary region are uncertain or inconsistent rules and we cannot classify the objects with certainty by using these inconsistent rules. The decision rules can have two conditional

probabilities associated with them - certainty and coverage which are closely related to the fundamental concepts of lower and upper approximations (Peters & Skowron, 2002).

Before generating the rules, one has to remove the redundant attributes from the information system and determine the minimal subset of attributes (called reducts) that are important for decision making by analyzing data dependency. It is simple to identify the redundant or indispensable attributes by using the concept of indiscernibility or equivalent relation. If a set of attributes and its superset correspond to the same discernibility relation, then any attribute that is a member of the superset but not to the set is considered as redundant (Pawlak, Grzymala-Busse, Slowinski, & Ziarko, 1995). The major applications of rough set in data mining include decision rule induction from attribute value tables and data filtration by template generation (L. Polkowski & A. Skowron, 1998; Skowron, 1995).

2.4.2 Decision Tree (ID3) and Rule Extraction

A decision tree is a machine learning technique based on constructing a decision tree. The concept for decision tree was popularized by Quinlan with ID3, *Interactive Dichotomizer 3* (Quinlan, 1986). ID3 develops a decision tree based on the greedy algorithm of entropy reduction to classify objects and make predictions for discrete class intervals. ID3 prunes a search tree based on the

entropy. Classification is based on recursive partitioning of the data set into categories involving intersection among the variables in various values. At each node of the decision tree, the remaining variables with highest reduction in entropy, or highest information gain, would be selected for the next stage of partitioning.

Decision trees do not require any prior knowledge of the data distribution and they work well on noisy data (Sushmita Mitra & Acharya, 2003). They reduce a volume of data by generating a fewer number of compact rules, while preservers the essential characteristics and accuracy. They determine whether the data has well-separated classes of objects. The most important feature of decision tree is their capability of breaking down a complex decision making process into a series of simpler decisions that are easily interpretable (Safavian & Landgrebe, 1991).

2.4.3 Neural Networks and Rule Extraction

A neural network is used to build an intelligent system based on the model that simulates the working network of simple processing elements or neurons in human brain (Hopfield, 1982; Thangavel & Pethalakshmi, 2009). It is believed to be responsible for the intelligence and discriminating power of humans. A neuron is made up of several protrusions called dendrites and a long-branch called the axon. Millions of neurons are linked together through the dendrites in

a massively parallel manner. The dendrites of neurons meet to form synapses where the message pass, and the neurons receive the pulses via the synapses.

When a neuron receives a set of input pulses, internal processes take place such as activation of neurons, and then the neuron sends out another pulse that is a function of the input pulses. Suppose the inputs x_1, x_2, ..., x_n are coming to the neuron and each input x_i is multiplied by its corresponding weight w_i, then the product $w_i x_i$ is fed to the neuron. The weight w_i represents the biological synaptic strength in a natural neuron. The neuron adds up all the weighted inputs as follows:

$$net = \sum_{i=1}^{N} w_i x_i$$

Finally, the neuron computes its output as a function of net, i.e. y = f(net) where f is called the activation or transfer function. The activation functions depend on the characteristics of applications. The neuron is considered a black box that receives input vector x and sends out a scalar output y. The same output y can be sent out through the multiple dendrites emerging from the neuron. Artificial neural networks can be viewed as a weighted directed graph where artificial neurons are nodes and directed edges are connections between neuron outputs and neuron inputs.

Though the current neural networks are far from achieving the real intelligence, as it was predicted, they have several real life applications that include pattern classification, clustering, optimization, and forecasting. A neural network can learn from the "environments" by employing one of the three paradigms of learning: supervised, unsupervised, and reinforcement. In supervised learning, each network output is compared against the desired output for each input. In unsupervised learning, neural networks are given some general guidelines and then learn by themselves. No specific input-output comparisons are made in this paradigm of leaning. Instead the network is tuned to certain criteria or algorithms to form categories (partition) by optimizing with respect to some independent parameters of the network (e.g. global energy). Reinforcement can be viewed as a special case of supervised learning and it learns the input-output mapping by trial and error while maximizing a performance index called a reinforce signal (Sushmita Mitra & Acharya, 2003).

2.4.4 Rough Set theory and Decision Trees (ID3)

Both rough sets and decision trees require data discretization since they cannot deal with continuous data. However other decision trees, such as CART or C4.5, are able to deal with continuous data. Neither rough sets nor ID3 requires any prior knowledge of the data distribution. ID3 may be more efficient than rough sets when the number of rules is very high, but it may overlook useful rules. Another difference is the way to represent knowledge or rules: rough set theory

develops an information table, while ID3 uses decision trees. Rough set may be selected for the problems that are better represented by tables than trees. On the other hand, ID3 is a right choice when the problems are better represented by a tree. In general, a tree data structure is efficient for searching and very inefficient for merging. Tables are easy to merge but difficult to search for information. A comprehensive comparison of rough set and decision tree (ID3) is provided in (Beynon & Peel, 2001; Daubie, Levecq, & Meskens, 2002; Mak & Munakata, 2002).

2.4.5 Rough Set Theory and Neural Networks

The major advantage of neural networks is their capability of parallelism with ease since each neuron can work independently. It can perform, with some degradation in service, even when a part of the network is damaged. It can deal with linear as well as nonlinear problems that are difficult to solve mathematically. Once we train a neural network, it can deal with new patterns which are similar to learned patterns. However, its weights have no direct meaning to us, and we cannot extract the underlying rules that may be generated from the neural network. Some research has been done on this issue, but no satisfactory solution has obtained yet (Munakata, 2008). Neural networks are also criticized for their poor scalability, longer computation time and occasional inconvergency. But once they converge, they can be copied to other systems and significant benefits can be obtained. Compare to neural networks, rule extraction is relatively easy for rough sets. Rule filtration is also more efficient in rough sets

compared to neural networks. It is also shown that the neural network performs best in robustness (90%) but its prediction ability is slightly worse compared to rough sets and ID3 (Mak & Munakata, 2002).

2.4.6 Synopsis of Existing Soft Computing Approaches for Pattern Generations (RS, DT, and NN)

Table 2. 3: An Overview of Soft Computing Approaches: RS, DT, NN

	Rough Sets (RS)	Decision Tree (DT)	Neural Networks (NN)
Deductive System	No	No	No
Inductive System	Yes	Yes	Yes
Primarily Numeric Data	No	No	Yes
Numeric and Descriptive Data	Yes	Yes	No
Data Filtration	Easy	Easy	Difficult
Ease of Rule Extraction	Yes	Yes	No
Rule Comprehensibility	Simple	Simple	Complicated
Training Time	Shorter	Shorter	Longer
Convergent	Yes	Yes	Not Always
Robustness	high	Higher	Highest
Classification Accuracy	Slightly Higher	Slightly Higher	High

An overview of the existing soft computing approaches for pattern generation - such as, rough sets, decision tree (ID3), and neural networks - is shown in Table 2.3. Rough sets, decision trees, and neural networks are all examples of inductive systems, where rules are induced by the systems rather than by the experts, unlike deductive systems, such as chaos and fuzzy systems, where rules are provided by the experts and output is determined by applying appropriate rules

for inputs. Primary inputs, intermediate data, and outputs are numeric for neural networks, whereas data can be high level description (e.g. very high, high, or low) or numeric (10 yrs) in rough sets or decision trees. Thus, if the data is in non-numeric form it must be converted to numeric before we can use it in neural networks.

Rule extraction and data distillation in rough sets and ID3 are easily compared to neural networks. The training phase of neural networks is much longer compared to rough sets and ID3. Moreover, neural networks may fail to converge sometimes when the data are inconsistent or incomplete. However, neural networks are much more robust compared to rough sets and ID3. Thus, if the data type is numeric and the objective of the data analysis is robustness, then neural networks are the best option in modeling training data. On the other hand, if the data type is qualitative (nonnumeric) and the objective is to obtain an easy-to-use decision table, then rough sets or ID3 is the better options.

In general, it is difficult to compare the two attribute-based learning techniques: rough sets and ID3 (Mak & Munakata, 2002). However, rough set theory is preferred as a rule induction approach in the context of the current research for the following reasons:

- Although, the tree structure of ID3 appears to be easy to understand, the rules or the patterns can be very complex and difficult to manipulate if we have

many attributes with numerous modalities (Daubie et al., 2002). Thus, ID3 is not appropriate in WSN, given that the sensor network problems are characterized by a large number of variables or sensors that produce continuous data (Kargupta, 2007).

- RS is well known for its feature reduction capabilities, which is important for high dimensional feature-oriented WSN (Pawlak, 2004). This feature is extremely important to many real world systems that exhibit non-polynomial complexity with respect to attribute dimensionality. An example includes a large scale water treatment plant that may require a huge number of attributes to monitor through sensors to perform diagnostic detection of faults ((Shen & Chouchoulas, 2000).

- Trees are extremely vulnerable to packet loss, which is often the case with WSN because of the unreliable transmission protocol. If the packet sent to the parent is lost, then the information to the entire subtree will be lost. Thus packet loss near the base station or sink can lead to disaster (Kargupta, 2007).

- Merging tables in RS for a knowledge base may be easier than merging trees in ID3 (Munakata, 2008). This feature of rough sets can be extremely useful while developing concurrent decision systems by transferring the RS rules to PetriNet, discussed in chapter five.

- The rules derived from RS are more extensive, while the rule generation by ID3 focuses on important rules based on the entropy measure (Munakata, 2008). This feature of RS can be very useful for some WSN applications, such

as forest fire detection when a large number of sensors fall in coverhole regions and are scorched in fire. The rules can be generated from the remaining sensors by employing RS theory.

- Rule generation by ID3 may take longer computation time compared to RS, since ID3 is based on entropy, a concrete quantitative measure in information theory. This is an important issue for highly resource-constraint WSN.

- There is no algorithm in decision trees that supports multivalued decision systems. However, there are well known algorithms in rough sets for multi-valued decision systems, which support the existence of multiple patterns that are generated at the same interval of time, and provides the association rule generation techniques for those patterns (Rzasa, Paluch, & Suraj, 2004). This is an essential feature of RS in the context of current research.

- Another important deciding factor is RS's inherent uncertainty handling capacity, given that the main focus of this research is uncertainty management in WSN.

2.5 Uncertainty Management in WSN

Uncertainty is an important measure in the analysis of risk. The problem of characterizing uncertainty in complex systems (e.g. sensor networks) is inherently interdisciplinary that requires in-depth knowledge of philosophical foundations of uncertainty, conceptual frameworks underlying the uncertainty

analysis, and technical methods to realize the uncertainty analysis (Ayyub, 2003). Besides, it is difficult to completely capture the nature of uncertainty and cover all its aspects because of its complex nature and its propagation through all epistemological levels of a system by varying degrees (Ayyub & Klir, 2006).

The uncertainties in sensor data streams may stem from missing data or unreliable data. Missing data may arise during sensor reading, format conversion, data discretization, data aggregation, data routing, data savings to storage devices, incorrect data labeling, etc. Unreliable data is either random or systematic. Random noise is an intrinsic part of data, caused by the movement of electrons or incorrect deployment of sensor nodes (Kargupta, 2007). Systematic errors can result from actuator uncertainty, sensor status uncertainty, limited sensing ranges, compromised nodes, improper channels, transmission collisions, routing uncertainty, and resource uncertainty (Y. Liu & Das, 2006).

2.5.1 Taxonomy of Uncertainty

Philosophers defined the nature and methods of acquiring knowledge and that knowledge evolved over time and produces different schools of thought. Uncertainty can be defined as "inherent deficiencies with acquired knowledge" (Ayyub & Klir, 2006). B. M. Ayyub's classification of uncertainty is shown in Figure 2.1. Philosophically, uncertainty can be broadly classified into objective (aleatory) uncertainty and subjective (epistemic) uncertainty. Some events or

variables are inherently random and nondeterministic in nature. This type of uncertainty cannot be reduced by increasing the knowledge and are called aleatory uncertainty. On the other hand, epistemic uncertainty stems from a lack of complete knowledge. Epistemic uncertainty can be reduced at the cost of increased resources, and this is the most common type of uncertainty in risk management. These two types of uncertainty can be combined together. However, this classification is not enough to define all aspects of uncertainty.
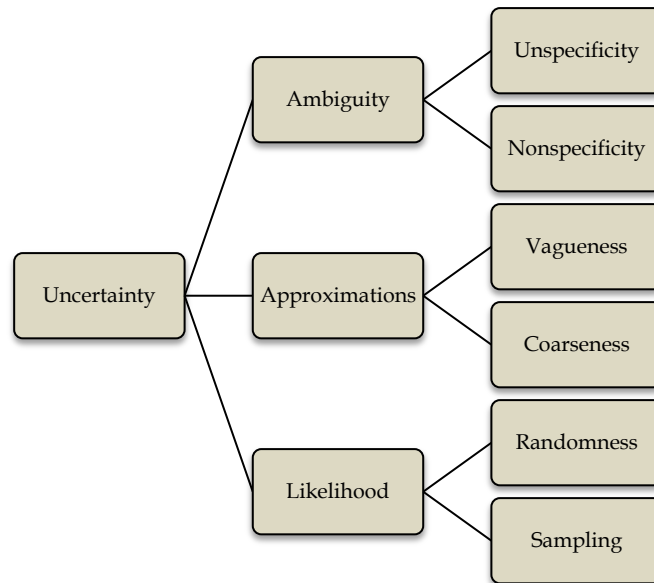
Figure 2. 1: B. M Ayyub's Classification of Uncertainty

Uncertainty can be classified based on its sources into three categories, such as ambiguity, approximations, and likelihood. In general, ambiguity and likelihood types of uncertainties in predicting the behavior and designing engineering systems are addressed by probability, statistics, and Bayesian methods. Probability distribution functions are used to model the uncertain parameters of

the system. Probabilistic methods that are useful for modeling this uncertainty include reliability methods, probabilistic engineering mechanics, stochastic finite element methods, etc. However, the axioms of probability and statistics are limited for the proper modeling, and analyzing the uncertainty arises from approximations. Uniform and triangular probability distributions are used to model some uncertainty parameters. The Bayesian approach is also another way to deal with this type of uncertainty by combining empirical and subjective information about the parameters (Ayyub & Klir, 2006). However, fuzzy set theory or rough set theory may be more appropriate to deal with this type of uncertainty as they are inherently more tolerant of imprecision.

Ambiguity arises from the possibility of multiple outcomes in outcome space (Ayyub & Klir, 2006). If the list of possible outcomes does not include all the outcomes from the outcome space, *unspecificity* will result. If the outcomes are not properly defined, *nonspecificity* will arise. Likelihood involves chances and gaming. Likelihood leads to randomness and sampling. Randomness is the result of non-predictability of outcomes, while sampling stems from using samples from populations.

Approximation arises from human cognition and intelligence, and thus the uncertainty that stems from approximations is subjective. It may result from the use of vague semantics in language, approximate reasoning, and removing

irrelevant components. Thus approximation results in vagueness, coarseness, and simplification. Vagueness arises from the imprecision of the membership of the elements to a set (Zadeh, 1965). Coarseness in information results from approximating an unknown relationship or a set by partitioning the universal space with the associated belief level for the partitioning subsets in representing the unknown relationship or set (Pawlak, 1992b). This approximation is addressed in rough set theory (Pal & Skowron, 1999; Pawlak, 1982).

In developing engineering models, simplifying assumptions are common practice for making the systems tractable. Errors resulted from these simplifications can be addressed by introducing biased random variables that are assessed empirically. Simplifications in systems can be also caused by using knowledge-based if –then rules to represents behavior based on fuzzy logic and approximate reasoning (Zadeh, 1965, 1975).

2.5.2 Formalisms of Uncertainty Measurement

Several formal and informal systems have been developed to characterize and model uncertainty in order to reason under uncertainty. Historically, the first approach for describing uncertainty has been the probabilistic method, in which the uncertainty is described by probability measures. The probabilistic approach works well in many engineering and scientific applications where we can determine the probabilities by observing the frequencies of the events or using

expert judgments (Nguyen, Kreinvich, & Dhompongsa, 2007). However, a problem arises when there is not enough statistics to determine the probabilities (i,e. instead of frequencies, we have the intervals of possible values of the quantity) or when the expert judgments are expressed in imprecise terms, such as "high" , "low", etc. Fuzzy set theory, proposed by Lotfi Zadeh (Zadeh, 1965), was one of the first formalisms to describe theses imprecise sets or fuzzy sets. Later in 1982, interval-uncertainty was addressed by Z. Pawlak by developing a new formalism called rough set theory (Pawlak, 1982). In the mean time, "A Mathematical Theory of Evidence" or Dempster-Shafer theory was published by Shafer in 1976 (Shafer, 1976) after reformulating the work of Dempster, which was a generalization of Bayes' theorem (Dempster, 1967, 1968).

All of these formalisms are well defined and well studied in various applications. Each of these formalisms can address certain types of uncertainties. However, the main challenge is to develop a mathematical formalism by combining several formalisms so that the uncertainty handling mechanism does need to be confined to a particular formalism to address real-life uncertainties that are often a combination of several types of uncertainties (Nguyen et al., 2007).

2.5.3 Probability Theory of Uncertainty

The concept of probability has its root in games of chance, where probabilities are computed based on the repetitions of an experiment and counting the

number of outcomes in an outcome space. This is considered a relative frequency based probability, which is a ratio of the number of occurrences of an event by the total number of repetitions. However, many real-world problems do not involve large numbers of repetitions, such as the probability of a satisfactory or unsatisfactory performance of a machine. To compute the probability for such systems, the idea of subjective probability came into the picture. The subjective probability or judgmental probability depends on the nature of the underlying event. The subjective probability is expected to reflect our knowledge about the systems regarding the occurrences of the events of interest. The events of satisfactory performance and unsatisfactory performance are mutually exclusive and collectively exhaustive in the universal outcome space. Thus subjective probability can be associated with degrees of belief and can form a basis for Bayesian methods (Ayyub & Klir, 2006; Ayyub & McCuen, 2003).

An axiomatic definition of probability is commonly given in the literature (Ayyub & McCuen, 2003; Lindley, 1975). The probability P should satisfy three axioms, which govern the basic uncertainty measures. The first axiom of probability states that the probability of an event A, which belongs to the set of all possible outcomes of the system (i.e. universal set X) is inclusively between 0 and 1 such that

$$0 \leq P(A) \leq 1 \qquad\qquad 2.4$$

The second axiom is derived from the definition of the universal set X, the set of all possible outcomes. The second axiom, the additive axiom, combines the probability of all events in universal set X, such that

$$\sum_{i=0}^{n} P(A_i) = 1. \qquad\qquad 2.5$$

If the probability of the universal set does not equal to 1, the universal set was not defined properly.

The third axiom, the axiom of multiplication, states that the occurrence probability of the union of mutually exclusive events is the sum of their individual occurrence probabilities such that

$$P(A_1 \cup A_2 \cup \ldots \cup A_n) = \sum_{i=0}^{n} P(A_i)$$

where, $A_1, A_2, \ldots, A_n$ are mutually exclusive events.

The third axiom is the basis of probability theory. Two important concepts, the relative frequency and subjective probability, meet this definition of probability and they are used as major tools to deal with uncertainty and model uncertainty. By increasing the number of repetitions, we can improve the estimate of the *true probability* in case of relative frequency at the cost of resources. However, a true probability may not exist specially for subjective probability. However, they provide a consistent, systematic, and robust framework for uncertainty management and decision making (Ayyub & Klir, 2006).

It is common to encounter problems in real-life that are associated with both objective (or empirical) and subjective (e.g. experience, intuition) types of information. It is wise to utilize both types of information and derive the probability that can be useful in uncertainty management and decision making. Bayesian probability can be calculated by assuming the subjective probability as a prior knowledge and frequency-based probability as objective information, and combining these two types into posteriori knowledge. The combination can be achieved by employing Bayes' theorem (Ayyub & McCuen, 2003; I. U. Sikder, 2003). If $A \subset X$ represents the objective information or event and B_1, B_2, …, B_n represent the prior or subjective information, we can compute the probability of an event by using the theorem of total probability as follows:

$$P(A) = P(B_1)P(A \mid B_1) + P(B_2)P(A \mid B_2) + …+P(B_n)P(A \mid B_n)$$

2.5

where, $P(B_i)$ is the probability of the event B_i and $A \mid B_i$ is the occurrence of A given B_i for i = 1, 2, …, n. This theorem is very useful to calculate the probability from the probabilities of the partitioning events and conditional probabilities, where it is difficult to calculate the probability directly (Ayyub & Klir, 2006). Bayes' theorem can be used to compute the posterior probability as follows:

$$P(B_i|A) = \frac{P(B_i)P(A|B_i)}{P(B_1)P(A|B_1)+ P(B_2)P(A|B_2)+…P(B_n)P(A|B_n)}$$
2.6

Where, the denominator represents the $P(A)$, $P(B_i)$ denotes the prior knowledge and $P(B_i|A)$ represents the posterior knowledge.

2.5.4 Fuzzy Set Theory of Uncertainty

Fuzzy sets were introduced in 1965 by Lotfi Zadeh to represent vagueness or fuzziness in real life data. Fuzzy set theory is a way of capturing the vagueness present in the real world, which was difficult to achieve by using traditional set theory. Fuzzy logic is not fuzzy but a precise logic of imprecise information and approximate reasoning. It provides an approximate but reasonable solution for systems that are either too complex or ill-defined for precise mathematical solutions (Zadeh, 1975, 1978).

People have started to recognize that objects, handwriting, voice, images, and other patterns are often distorted, incomplete, and fuzzy, and thus a pattern should be allowed to have membership to more than one class. For instance, a patient with a certain set of symptoms can be simultaneously suffering from multiple diseases and the symptoms are not always numeric, such as low, high, very high, etc. Fuzzy concept was developed to explain this situation.

Often the logic behind human reasoning is not the traditional two-valued or multivalued logic, but logic with fuzzy truths and fuzzy rules of inference (Sushmita Mitra & Acharya, 2003). In classical set theory, the elements either fully belong to the set (i.e. membership of 1) or not at all (i.e. a membership of 0). Fuzzy set theory relaxes this restriction by associating any value in the range of 0

and 1 (0 ≤ degree ≤ 1) as a membership for each element. The membership value shows the degree of compatibility or similarity of an event with an imprecise concept representing a fuzzy set, while the probability of an event is based on the frequency of the occurrences. A fuzzy set can be defined as a set of ordered pair $A = \{x, \mu_A(x)|\ x \in U\}$. The function $\mu_A(x)$ is called the membership function for A, mapping each element of the universe U to a membership degree in the range [0, 1]. Fuzzy membership function $\mu_A(x)$ has the following properties:

- $\mu_{U-A}(x) = 1 - \mu_A(x)$ for any $x \in U$

- $\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x))$ for any $x \in U$

- $\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x))$ for any $x \in U$

This shows that the membership of an element to the union (or intersection) of sets is uniquely determined by its memberships in the constituent sets. This is an important property of fuzzy set in theoretical as well as practical aspects since this property allows simple set operations on fuzzy sets (Pawlak, 2004).

An example of a fuzzy set can be "All tall students in the class". Obviously "tallness" is not a step function from 1 to 0 at a certain height, say 6′. Thus it would be natural to associate a degree of tallness to each element of the fuzzy set. In extreme cases, if the degree is zero the student does not belong to the set and if the degree is 1 the student belong 100 % to the set of tall students. Besides, it is context dependent, since tallness depends on the student's gender, ethnicity, etc.

Fuzzy set theory is well recognized in reasoning under uncertainties stems from deficiencies of information and in decision making under risks, subjective judgment, vagueness, and ambiguity. The deficiency results from incomplete, imprecise, vague, or contradictory information. As a generalized theory of classical set theory, fuzzy sets theory offers greater flexibility to handle uncertainty issues.

2.5.5 Dempster-Shafer Theory of Uncertainty

Evidence theory, also called Dempster-Shafer theory was introduced by A. P. Dempster (Dempster, 1967, 1968) and further developed by G. Shafer (Shafer, 1976). It is a generalization of probability theory where the sample space of probability theory corresponds to the frame of discernment in evidence theory. In probability theory, unlike evidence theory, a probability is assigned to each element of a sample space and the probability of any event or a subset of the sample space can be computed by Kolmogorov's addition axiom in probability theory (Kolmogorov, 1956). However the basic idea of evidence theory is a basic probability assignment, where function $m: 2^\theta \rightarrow [0, 1]$ such that

$$m(\emptyset) = 0$$

$$\sum_{X \subseteq \theta} m(X) = 1$$

m(X) is called the *basic probability number* of X. Thus in evidence theory, basic probability numbers are assigned to the members of the subsets of the frame of

discernment, where in probability theory probabilities are assigned to singleton sets. In evidence theory, it is possible to have the situation when $m(\{x\}) \neq 0$ and $m(X) = 0$ for $x \in X$. However in probability theory if $p(x) \neq 0$ then $P(X) \neq 0$. A subset X of the frame of discernment is called a focal element if $m(X) > 0$.

There are two commonly used functions in evidence theory, a *belief function* and a *plausibility function*. A *belief function Bel*: $2^{\theta} \rightarrow [0,1]$ over Θ is defined as follows:

$$Bel(A) = \sum_{B \subseteq A} m(B)$$

A *plausibility function Pl*: $2^{\theta} \rightarrow [0,1]$ over Θ is defined as follows:

$$Pl(A) = \sum_{B \cap A \neq \emptyset} m(B)$$

The two measures are related to each other as follows:

$$Pl(A) = 1 - Bel(\overline{A})$$

There are certain types of uncertainties that cannot be classified by traditional probability theory, such as nonspecificity that stems from imprecision associated with the sizes or cardinalities of relevant sets of alternatives. For example, if we have multiple patterns occurring at the same interval and there is no specific choice, then nonspecificity arises. A basic probability can be assigned to each alternative of the template sets by employing Demster-Shafer theory of evidence, and nonspecificity can be computed by using these probability assignments and extending the Hartley function.

2.5.6 Rough Set Theory of Uncertainty

Although rough set theory has been developed to deal with uncertainty and imprecision where there are some overlap with other formalisms such as fuzzy set theory (Dubois & Prade, 1990), evidence theory (Skowron & Grzymala-Busse, 1994) and statistics (Krusinska, Slowinski, & Stefanowski, 1992), it stands in its own right (Pawlak, 1992a).

Rough set theory has emerged as a major mathematical tool for managing uncertainty that arises from granularity in the domain of discourse or discernibility between objects in a set. The objective of this theory is to approximate a rough or imprecise concept in the domain of discourse by a pair of exact concepts, such as lower and upper approximations. These exact concepts are determined by an indiscernibility relation, developed from the attribute sets of the objects in the domain. The lower approximation is the set of objects that are certainty belong to the imprecise concept, while the upper approximation is the set of objects that possibly belong to the rough concept.

These two approximations are used to define the discernibility matrix, discernibility function, reducts, and the degree of dependency among the attributes – all of them help reduce data and manage uncertainty stems from data granularity and discernibility. The effectiveness of this theory has been studied in the domains of artificial intelligence and cognitive sciences for

representation and reasoning under vague and imprecise knowledge, data classification, data mining, and knowledge discovery (Slowinski, 1992).

Some important uncertainty measures that have been developed in rough set theory include a quality measure of lower approximation and a quality measure of upper approximation. For a given set X, the quality of lower approximation is the ratio of the number of all elements in the lower approximation of X to the total number of elements in the information tables. Similarly, the quality of upper approximation is the ratio of number of all elements in the upper approximation of X to the total number of elements in the information table (Pawlak et al., 1995). The details of RS formalism are provided in the following chapter.

2.5.7 Rough Set Theory and Probability Theory of Uncertainty

Both theories deal with similar problems such as reasoning under uncertainty about data; however their approaches are different. In general, some problems are solved better by traditional probability theory, while the others are well addressed by rough set theory. For instance, probability theory may work better when the data sample size is large and the data distribution is close to normal. On the other hand, for fewer sample size or non-normal distribution of dataset rough set may be a potential candidate since rough set, unlike traditional probability theory, does not make any priori assumptions of data size or its

normal distribution. However, there is no theory that determines exactly what circumstances which approach works better; they complement each other (Munakata, 2008).

Bayes' theorem is the essence of probability theory and statistical inference. It provides posterior distribution from the prior distribution, when combined with the evidence provided by the data. Rough set theory sheds new light on the Bayes' theorem to explore a new direction in traditional Bayesian data analysis. It does not involve prior or posterior probability, instead it provides some probabilistic structure of the data being analyzed. It is shown that any decision table satisfies total probability and Bayes' theorem. With the help of this property, it is possible to draw decisions and discover data patterns from data without any prior knowledge. The difference between the role of Bayes' theorem in statistical reasoning and its role in rough set theory is clearly demonstrated in a book by Z. Pawlak (Pawlak, 2004).

2.5.8 Rough Set Theory and Fuzzy Set Theory of Uncertainty

Fuzzy set can be considered as a specialization of rough sets (Jenssen, Komorowski, & Øhrn, 1998). However, there are basic differences between then them. Fuzzy sets represent vagueness of a quantity, such as obtaining linguistic quantities from experts, while the rough sets represent coarseness as an approximation of a crisp set. In fuzzy sets, the temperature of a sensor can be

both as high or very high, but with different membership values in the range of [0,1]. But in rough sets, the temperature of a sensor is either high or very high, but the cardinality of the sets of the sensors with high temperature and very high temperature are uncertain. Fuzzy sets are well suited for control system, while rough sets are well known for classification.

The comparison between rough set and fuzzy set theories can be explained with an example. Consider a group of people who have applied for an auto loan in 2008 as the universe of objects under consideration. The number of people who applied for auto loan in 2008 is a crisp set and there is no vagueness associated with this set. However, if we want to know "how many young people have applied for auto loan in 2008?" The answer is a fuzzy set since 'young' is a vague concept and the measure of youngness does not change abruptly and it ranges from 0 to 1. According to Fuzzy theory, we can associate a degree of youngness, ranging from 1 to 0, to each person who has applied for auto loan if we know the age of each applicant. People with 18 <age> 25 can be assigned 1, 25 < age < 30 can be assigned 0.8, 30 <age < 35 can be assigned 0.6, etc.

How we will count the number of young applicants for an auto loan if we do not know the age of each applicant? Using rough set theory we can determine the relevant attributes regarding age and estimate the age of each applicant. The useful information regarding a person's age in this context can be his or her year

of high school diploma, highest academic degree, work experience, current salary, marital status, number of children, etc. Before we determine the important attributes for decision making, we need to create a training set from previous applicants where the age of every people were known. The training set contains all possible information about the applicants, including their age. However, applicants do not always provide correct information – some information is missing, incorrect, or irrelevant. By employing rough set theory on the training set, we can determine the importance of attributes and their degree of dependency even when data is not precise or complete. After the training session, we can use the useful information for actual dataset (test dataset) and estimate the age of each applicant.

Fuzzy set can do the same task by developing many fuzzy if-then rules, but it would be time-consuming. Instead, it is recommended to use rough set as a front-end of a fuzzy system to estimate the age. In some problems where condition attributes are expressed in terms of fuzziness and fuzzy logic, fuzzy set can be used as a front-end of a rough set system. In this case, some condition attributes are preprocessed to create a smaller number of intermediate attribute values by using fuzzy logic. Thus, rough set and Fuzzy set can complement each other (Munakata, 2008). Fuzzy rough sets are rough sets derived from fuzzy partitions, whereas rough fuzzy sets are rough set approximations of fuzzy sets derived from crisp partitions (Ayyub & Klir, 2006).

2.5.9 Rough Set Theory and Dempster-Shafer Theory

Rough set theory and Dempter-Shafer theory of evidence are both well-known for dealing uncertain knowledge and approximate reasoning. However, Dampster-Shafer theory uses *belief* or *plausibility* functions as a major computing tool, while rough set theory uses lower and upper approximation sets to represent the relations among the attributes. However, Grzymala-Busse and Skowron suggested a clear connection between rough sets theory and evidence theory, and they also showed that a *belief function* in Dempster-Shafer theory can be computed from the quality of lower approximation in rough sets and the *plausibility function* in Dempster-Shafer theory corresponds to the quality of upper approximation in rough sets (Grzymala-Busse, 1991; Skowron & Grzymala-Busse, 1994). The following description is adopted from (Skowron & Grzymala-Busse, 1994).

Let $\Theta_A = \{1, 2, \dots, r(d)\}$ be the *frame discernment* defined by the decision d in the decision system S = (U, $A \cup \{d\}$), where U is a non-empty finite set of objects called the universe and A is a non-empty finite set of attributes and $d \notin A$ is the decision attribute. Each attribute a $\in A$ can be considered as a function that maps elements of U into a set V_a. V_a represents the value set of attributes, such that

$$a: U \rightarrow V_a$$

For any $\theta \in \Theta_A$ the following equality holds

$$Bel_A(\theta) = \frac{|\underline{A} \cup_{i \in \theta} X_i|}{|U|}$$ 2.1

It defines the relationship between the *belief function* $Bel_A(\theta)$ in Dempster-Shafer theory and the quality of the lower approximation in rough set theory. Thus the *belief function* $Bel_A(\theta)$ is defined as the ratio of the number of elements that are certainty classified into the union of $\cup_{i \in \theta} X_i$ to the number of elements in U. Similarly, for any $\theta \in \Theta_A$ the following equality holds

$$Pl_A(\theta) = \frac{|\overline{A} \cup_{i \in \theta} X_i|}{|U|}$$ 2.2

It defines the relationship between the *plausibility function* $Pl_A(\theta)$ in Dempster-Shafer theory and the quality of the upper approximation in rough set theory. Thus the *plausibility function* $Pl_A(\theta)$ is defined as the ratio of the number of elements that can be possibly classified into the union of $\cup_{i \in \theta} X_i$ to the number of elements in U.

Another difference between rough sets theory and Dempster-Shafer theory is that rough set theory is data-driven and objective but the Dempter-Shafer theory can be subjective when the basic probabilities are assigned by experts' judgments (Munakata, 2008). The qualities of the approximations in rough set theory are computed from the given information table since rough set theory is objective. On the other hand, the values of *belief* or *plausibility* are assumed to be provided by an expert in Dempster-Shafer theory as the theory is based on the subjective paradigm (Skowron & Grzymala-Busse, 1994). However, we can compute the

basic probability assignment in practical applications without any knowledge of subjective or expert judgments by employing the rough set based evidence theoretic approach mentioned above (equation 2.1 and 2.2).

2.5.10 Synopsis of Existing Uncertainty Handling Formalisms

Probability theory can be considered as a special case of the Dempster-Shafer theory of evidence. When all the focal elements for a given basic assignment, m, are singletons, the belief measure and the plausible measure merge into a single measure, which corresponds to a classical probability measure. The term singleton means that each subset A_i of the family A of subsets (i.e. evidence body), contains only one element. The differences between the evidence theory and probability theory (Ayyub & Klir, 2006) are as follows:

- By using a basic assignment in evidence theory, we can compute the belief and plausibility measures that map the power set of X to the range [0, 1].

- A probability assignment, such as a probability mass function in probability theory, maps the universal set X to the range [0, 1].

The concept of rough membership function, introduced by Pawlak (Pawlak, 1982), is to some extent comparable to fuzzy membership function. However, Pawlak argues that the concept of rough membership is wider than fuzzy membership because of the following reasons:

Rough membership function $\mu_A(x)$ has the following properties:

$$\mu_{A \cup B}(x) \geq \max(\mu_A(x), \mu_B(x)) \text{ for any } x \in U$$

$$\mu_{A \cap B}(x) \leq \min(\mu_A(x), \mu_B(x)) \text{ for any } x \in U$$

This shows that the rough membership of an element to the union (or intersection) of sets is uniquely determined by its memberships to the constituent sets. This is an important properly of rough set in theoretical as well as practical aspects since this property allows simple set operations on rough set. The same properties were shown previously for fuzzy set, where the corresponding relationships for fuzzy members are equalities. By comparing theses properties for rough sets and fuzzy sets, it is clear that rough membership can be regarded formally as a generalization of fuzzy membership (Jenssen et al., 1998). An overview of uncertainty management techniques based on different mathematical formalisms is provided in Table 2.4.

Table 2. 4: An Overview of Uncertainty Management Techniques

Author	Formalism	Description
Lakshmanan, Leone, Ross, & Subramanian (1997)	Probability Theory	Computed probabilities of complex events from the elementary events by considering the interdependencies of events, attributes, and tuples. They also developed a system called ProbView, based a single unified framework, for combining probabilities from known interdependencies.
Morrissey (1990)	Probability Theory	Proposed a method that estimates the uncertainty introduced by imprecise information and ranks objects for presentation to a user. Their uncertainty estimation is based on both self-information and entropy measures. However their representation did not consider interdependencies of events and subjective probability estimates as the part of attribute values.

Yager (2000)	Dempster-Shafer Evidence Theory	Focused on the development of multi-criteria decision function based tools that are capable of mimicking the complexity of human intelligence. They obtained a class of ordered aggregation functions from two important measures of Dempster-Shafer theory plausibility and belief. They also have shown how different components, such as value, a subset of criteria, and a list for combining the criteria, were combined for hierarchical decision making.
Zhou & Mourelatos (2008)	Dempster-Shafer Theory& Bayesian Approach	Proposed an optimization method based on the evidence theory which is capable of handling epistemic as well as random uncertainties. Instead of expert systems, they have used Bayesian approach to form the basic probability assignment for a specified confidence percentile, using only the available sample points within ranges. They demonstrated the merit of their optimization technique by examples.
Grzymala-Busse (2003)	Rough Set Theory	Described incompletely specified decision by introducing characteristic relations, which is reduced to an indiscernibility relation for completely specified decision table. The characteristic relations were computed by using an idea of block of attribute-value pairs, used in some rule induction methods (e.g.LEM2). They also provided several definitions of lower and upper approximations, which converge to traditional approximations in the absence of missing attribute values.
Düntsch & Gediga (1998)	Rough Set Theory	Proposed three approaches for prediction based on RS by using various entropy measures, excluding statistical entropy measure. They developed a procedure called SORE (Searching Optimal Rough Entropy Sets), based on the principle of indifference combined with the maximum entropy. They demonstrated the applicability of the proposed method by comparing its error rate with C4.5 for 14 published datasets.
Wu & Mendel (2007)	Fuzzy Set Theory	Defined and derived formulas for computing four uncertainty measures of interval type-2 fuzzy sets (IT2 FSs): cardinality, fuzziness, variance, and skewness. Definitions are based on Representation theorem (Mendel –John representation) for IT2 FSs. These measures could be useful in designing IT2 fuzzy logic systems based on principle of uncertainty and measuring the similarity between two IT2 FSs.
Ganoulis (2007)	Fuzzy Set Theory & Partial Differential Equation	Proposed a method using fuzzy set theory, in combination with partial differential equations, to propagate uncertainties and estimate the risk of environmental water pollution. Uncertainties in input variables and values of the model parameters are first introduced as fuzzy numbers and then they are propagated using fuzzy arithmetic. The output

		variables, such as water pollution and environment risk, are estimated in terms of fuzzy numbers. They have used well-known advection-dispersion mathematical model for simulation of environmental water quality.

The literature review of uncertainty management formalisms reveals the following information:

- Probability theory is a special case of Damster-Shafer theory of evidence (Ayyub & Klir, 2006).

- Rough membership is a generalization of fuzzy membership (Jenssen et al., 1998; Pawlak, 1982).

- The belief/plausibility function in Dempster-Shafer evidence theory can be computed from the quality of lower/upper approximation in rough set theory (Grzymala-Busse, 1991; Skowron & Grzymala-Busse, 1994).

However, all these formalisms are well developed and thoroughly investigated. The selection of the formalisms depends on the application domain and the type of uncertainties that need to be addressed. Often a combination of formalisms is the solution instead of confining to a single formalism.

2.5.11 Recent Work on Uncertainty Management in WSN

Research related to uncertainty issues in wireless sensor networks addresses broadly two distinct aspects: location or deployment uncertainty and data information uncertainty resulting from data aggregation. Location uncertainty

emerges when the placement of sensors is required in a sensor field but the exact locations of the sensors are not known. From the viewpoint of location uncertainty, routing and location protocols have been proposed for events reporting to mobile sink or target tracking (Howard, Matari´c, & Sukhatme, 2001; Patwari & Hero, 2003; Zou & Chakrabarty, 2004). Zou & Chakrabarty (2004) developed a model to optimize the number of sensors and their location in a distributed sensor network.

Wang, Yip, Yao, & Estrain (2004) propose a Bayesian method to describe the lower bound of localization uncertainty in terms of minimum entropy in sensor networks. The dependency of localization uncertainty on the sensor network topology is determined by using the Bayesian method and the Cramer-Rao bound (CRB). Thus the algorithm identifies the region where the target is relatively located with some accuracy by assuming Gaussian sensing uncertainty. However, the model did not consider heterogeneous sensors and non-Gaussian sensing.

Buttyán, Schaffer, & Vajda (2006) propose RANBER, an algorithm for resilient data aggregation in sensor networks by eliminating outliers, based on the well-known RANSAC paradigm. The RANBER algorithm is useful even when a large percent of the sample has been compromised by an attacker. The model consists of an aggregator function and a detection algorithm. The detection algorithm

analyzes the input data before the aggregation function is called and detects unexpected deviations in the received sensor readings. The sample is divided into two halves and the sum for each half is calculated. If the difference of the two sums is greater than a threshold value, it indicates an attack.

Reznik & Kreinovich (2004) investigate the issues for improving the reliability, accuracy, and uncertainty management of the decisions based on the application of the meta-level models in sensor networks. The meta-level model represents a relationship or association between different sensors. The model depends on expert opinion, data mining techniques (genetic algorithm, neural networks, decision trees), and the type of data collected from sensors. The model attempts to integrate sensor results with the association information available at aggregation nodes and considers both neuro-fuzzy and probabilistic methods to review sensor results and association information.

From a database point of view, Cheng & Prabhakar (2003) introduce a data uncertainty framework that represents different levels of uncertainty in information. Depending on the amount of uncertainty in information given to the application, different levels of imprecision are presented in a query answer. They examine the situations when query answer imprecision can be represented qualitatively and quantitatively. An application of range query in a sensor network requires handling interval query and management of uncertainty

intervals qualitatively; however, the use of other queries, such as nearest-neighbor queries requires probabilistic threshold information.

2.5.12 Critical Summary of Literature Review

Existing data stream mining or data aggregation based routing algorithms in WSN barely address one important dimension in the optimization space for data routing and data aggregation, namely uncertainty. Some recent work shows some initiations in the area of location uncertainty; however the aspect of data uncertainty is largely unexplored. Given the growing demands for complex domain specific applications of sensor networks such as object tracking and event identification, it is imperative to deal with the uncertainty and the uncertainty propagation through data aggregation in a real time environment. Therefore, uncertainty - combined with energy, power, bandwidth, and network life time - should be incorporated in optimization model of aggregation-driven routing protocols for wireless sensor networks.

There are numerous types of uncertainties, already identified, and several well established mathematical formalisms to quantify and manipulate these types of uncertainties. However, they need to be reinvestigated and readdressed in the context of WSN. In general, probability theory is well recognized for handling uncertainties caused by random components. Fuzzy set theory may be used to manage vague concepts (e.g linguistic attributes), while Dempster-Shafer

evidence theory can be employed to handle the uncertainty due to information incompleteness. Rough set theory can be a potential choice to address uncertainty when it stems from coarseness. However, there is no unified mathematical formalism that integrates all existing formalisms and addresses real-life uncertainties in wireless sensor networks which are often a combination of several type of uncertainties (Nguyen et al., 2007).

The five important steps of our uncertainty management scheme are uncovered from our extensive literature review (Ayyub & Klir, 2006; Klir & Folger, 1988):

- Identify the type of uncertainties in the context of the proposed data aggregation scheme in WSN.

- Find an appropriate mathematical representation of each of the identified uncertainty type.

- Develop a calculus for each of the uncertainty types so that it can be quantified and manipulated.

- Determine a way to measure the uncertainty to the given context of WSN and develop a research methodology where all identified types of uncertainties are addressed.

CHAPTER III
METHODOLOGICAL FOUNDATIONS OF PATTERNS
GENERATIONS & UNCERTAINTY MANAGEMENT

3.1 Introduction

In the preceding chapter, the research questions have been transformed into a theoretical model, consisting of theoretical constructs, causal relationships and the measures. The theoretical model has developed based on the analysis of literature review. The selection of an appropriate research methodology is critical to the success of any research. This chapter describes a research methodology for discovering spatio-temporal patterns in sensor data streams and generating approximate rules by integrating rough set theory and stream data processing. The research methodology also describes uncertainty management issues associated with approximate rule generation in WSN by identifying several uncertainty measures and determining their tradeoffs in decision making. The

research methodology is developed in terms of philosophy, objectives, scope, assumptions, algorithms, and validation.

A sensor network gathers a huge amount of stream data from the environment, where most of the patterns generated from such data are obvious, redundant or uninteresting to the users. We need a technique for identifying the useful and interesting patterns that meet some user-defined threshold value so that only these interesting patterns can be reported to the sink node in a WSN. This technique has a potential to significantly reduce the amount of data communication in severely resource-constrained wireless sensor networks environment. However, the number of elements in data streams can be unbounded as sensors are collecting information for constantly evolving entities about the environment like temperature, pressure, etc. Thus, traditional data mining techniques are not appropriate to analyze such sensor data streams since it is not possible to store the entire data stream in a stream processing system.

One of the potential solutions can be to monitor and analyze sensor data streams immediately as they arrive and discover interesting patterns and generate the corresponding rules that represent the sensor data stream. Thus instead of storing the entire data stream, only the interesting patterns or rules can be stored on the processing system and the patterns or rules can be periodically sent to the sink. However, the number of patterns or rules generated from unbounded data

streams can be very large, which is neither feasible nor economical unless we use some form of approximation to reduce the number of rules. All uncertainty issues that stems from theses approximations need to be addressed when we employ these rules for prediction. Thus, it is important to develop an appropriate and well defined methodology that incorporates these vagueness and uncertainty measures in its model parameters and provides tractable, robust, and low-cost approximate solutions despite its imprecise rules or incomplete data set.

Granular computing is a unified framework for theories, methodologies and techniques that can be very useful in finding meaningful patterns in data by expressing and processing chunks of information – clumps of attribute values drawn together by indistinguishability, equivalence, proximity or functionality (Zadeh, 1978). Rough set theory is a popular mathematical framework for granular computing. Rough set is suitable for handling the issues related to understandability of patterns, uncertainty in data information, and it can provide approximate solution quickly. It is a major mathematical tool for handling uncertainty that arises from granularity in the domain of discourse or indiscernibility between objects. Soft granules can be obtained by membership functions or by lower and upper bounds. While increased granularity reduces attribute distinctness and results in the loss of useful information, finer granularity creates partitioning problem. Soft computing allows one to focus on

some specific and problem-oriented subsets of a complete database, resulting in modularization (Sushmita Mitra & Acharya, 2003).

The current research is framed in the direction of data uncertainty management in artificial intelligence, soft computing, and granular computing. Hence, the limitation of classical probability model is replaced by generalized imprecise probability model, where the scope of truth functionality of probabilistic statements is approximated in the range of upper and lower scopes.

3.2 Elements of Methodology

The necessary elements to define the methodology for an information system development include an underlying philosophy, a statement of its objective, a statement of its scope, premises and assumptions.

3.2.1 Philosophy

The philosophical foundation of the current research methodology is guided by the nature of the problem domain and the framework within which the phenomena is being carried out. There are two philosophical views of uncertainty: ontological and epistemological views. The current research does not address the ontological uncertainty that deals with the following questions:

where, when, and how densely to deploy the sensors? How many sensors should be used to optimize coverage areas? How frequently would the sensors report information about the features? How many features will a sensor sense? Instead, the current research focuses on the epistemological uncertainty that addresses the representational uncertainty in a dataset. By lowering the degree of precision in a dataset, data patterns can be more visible and identifiable. The current research methodology focuses solely on the structure of the given dataset instead of relying on the model parameters or external parameters, such as membership function or normal data distribution. The non-invasive approach (free from data distribution assumption) of the research is guided by the philosophy "let the data speak for itself."

3.2.2 Objectives

The methodological objective is to develop a theory for characterizing uncertainty and uncovering the tradeoffs among uncertainty measures in spatio-temporal patterns generated from sensors' data in the framework of rough sets, granular computing, and data stream processing. The operational objective is to test the effectiveness of the theory by employing it in unsupervised classification of spatio-temporal templates. The methodological foundation is also provided to identify redundant spatio-temporal patterns and select indispensible patterns for decision making from symbolic, quantitative, and outlier data by using a rough set approach.

3.2.3 Research Scope

Research is an organized and systematic approach to find the truth. It is always focused on relevant, useful, and important questions and limited to a specific scope. The scope of the current research is as follows:

- Sensors are in general static.

- Nodes are assumed to have limited local processing capabilities.

- Our model supports node heterogeneity.

- Our model is appropriate when data has some spatial and temporal correlations.

- The proposed methodology may not be useful when the signal changes very rapidly and does not generate enough tuples to create a temporal information system.

- Our research methodology is architecture independent.

3.2.4 Premises and Assumptions

- A fundamental assumption stems from the validity of the bias introduced in the methodology by selecting rough set formalism in our model.

- The second assumption is the closed world assumption – the universe of discourse U contains no other objects except those we have included in our information system.

- The third assumption is the fundamental assumption of rough set – we can associate some information (data, knowledge) with every object of the universe of discourse (Pawlak & Slowinski, 1994).

- The fourth assumption is that the variables are disjunctive variables that have a single value at any given time. Measures of uncertainty have been almost exclusively investigated in literature in terms of disjunctive variables (e.g. person's age, air pressure at a particular location). Probability theory, possibility theory, Dempster-Shafer theory, and several other theories of imprecision consider only disjunctive variables; they do not consider conjunctive variables (e.g. children of a person, courses taken by a student) (Ayyub & Klir, 2006).

3.3 Theoretical Foundation

It is important to understand and manipulate imperfect knowledge given that the sensor data is often imprecise, incomplete, ambiguous, and redundant due to the resource constraints of WSN, densely deployed inexpensive and error-prone sensors, and the unfriendly environment of their deployment. Often an approximate solution is the only viable solution, while precise solutions are either unfeasible or too expensive. When the information is incomplete, uncertain, or vague, and it is difficult to differentiate elements, it may be

convenient to consider granules, clumps or groups of indiscernible elements, for performing operations (Pal, Shankar, & Mitra, 2005).

Various soft computing methodologies, such as fuzzy logic (Zadeh, 1971), Decsion tree ID3 – Iterative Dichotomizer 3 (Dhar & Tuzhilin, 1993; Pao, 1989; Quinlan, 1986, 1992), neural networks (Fu, 1999), and rough set (Pawlak & Slowinski, 1994), have been applied to handle the challenges posed by uncertainties and provide approximate solutions. Each of them has a distinct methodology for addressing problems in its domain and providing an acceptable solution at a reasonably low cost by exploiting the tolerance for imprecision and uncertainty. For instance, fuzzy sets provide a framework for classifying uncertainty in complex problems by allowing gradual changes and descriptive expressions. However, the generic fuzzy set theory does not have learning capability, and its analysis is based on the fuzzy membership function (Pawlak, 1997).

ID3 is a decision analysis technique based on the greedy algorithm of entropy reduction in constructing the decision tree. ID3 prunes search tree based on the entropy. ID3 may be more efficient when the number of rules is very high, but it may overlook useful rules. Another difference is the way to represent knowledge or rules: rough set theory develops information tables, while ID3 uses decision

trees. A comprehensive comparison of rough set and decision tree (ID3) is provided in (Beynon & Peel, 2001; Daubie et al., 2002; Mak & Munakata, 2002).

Neural networks and rough sets are commonly used for classification of uncertainty and rule generation (Sushmita Mitra & Acharya, 2003). In general, the number of rules generated from a given dataset by using neural networks is much larger than the rough sets (Al-Qaheri, Hassanien, & Abraham, 2008; Mak & Munakata, 2002; Iftikhar U. Sikder & Munakata, 2009). Besides, it is often difficult to explain how the data patterns are generated in neural networks because of the complexity and nonlinear data transformation taken place in multiple hidden layers. Another disadvantage of neural networks is that the rule extraction and filtration are less efficient, compared to rough sets. (Mak & Munakata, 2002).

3.3.1 Rough Set Theory

The major objective of rough set theory is to generate rules from complex data by removing the features that are not important for decision making. The theory is based on the assumption that every object in the universe is associated with some information (Pawlak & Slowinski, 1994). The decision table generated, based on the important attributes, can be ready used in solving multi-attribute decision problems. Rough set theory has been successfully applied in medical

diagnosis (Tsumoto, 1999), industrial control (Munakata & Pawlak, 1996), and marketing analysis (Kowalczyk & Piasta, 2006).

3.3.1.1 Introduction of RST

Rough set theory (RST), proposed by Zdzislaw Pawlak in 1982, has been significantly enhanced by a number of researchers and practitioners (Banerjee, Mitra, & Pal, 1998; Pawlak et al., 1995; Skowron & Polkowski, 1998). It has been widely used in knowledge discovery, data mining and approximate reasoning when data set is incomplete or imprecise. The main idea is the classification of empirical data by selecting the degree of roughness or precision of data and making subsequent decisions. The philosophy of rough set theory is to let the data speak for itself. Very few assumptions are made about the data. Attributes require only some notion of inequality defined on their domains.

The main advantage of rough set is that it is inherently data driven and "noninvasive" (L. Polkowski & A. Skowron, 1998). It does not require any additional information about data, like probability in statistics, basic probability assignment in Dempster-Shafer theory, or degree of membership in fuzzy set theory. By utilizing the structure of the given data from sensor networks, it is possible to develop the numerical value of imprecision or a membership function without requiring any subjective inference on distribution function.

The theory is based on two important ideas: the indiscernibility relation that describes indistinguishable objects and the concepts that are represented by lower and upper approximations (I. U. Sikder, 2003). Rough sets allow multiple memberships to deal with indiscernibility, while fuzzy set uses partial membership to deal with uncertainties. Applications of RST include broad spectrum of areas: bioinformatics, engineering, finance, marketing and music.

3.3.1.2 Selection of RST for Analyzing Sensor Data

- Rough set theory is inherently data driven, "non-invasive", and application independent. It is based on the philosophy "let the data speaks for itself." Thus, rough set based methodology developed for one sensor network application can be used for another sensor network application with a minor or no modification at all.

- Unlike fuzzy set theory or statistical analysis, a unique advantage of a rough set is that it does not rely on additional model assumptions or external parameters. It does not require membership function. This is an important property for the selection of rough set as a sensor data mining tool.

- Rough sets may be a better option than statistical analysis (e.g. discriminant analysis) when the underlying data distribution significantly deviates from a normal distribution since RST does not make assumptions about statistical distributions of data (Stefanowski, 1992). In wireless sensor networks, sensors are often deployed in harsh and unfriendly environments, such as terrains or

battle-fields where a large number of sensors are inaccessible and the only data collected by the accessible sensors may not provide a normal distribution.

- Rough sets can be more efficient than statistical analysis when the sample size is too small to define a data distribution (Stefanowski, 1992). This feature of RST is very useful in sensor networks when a large number of sensors fall into the cover hole regions and are destroyed by forest fires.

- Rough sets are inherently equipped to handle inconsistency and ambiguity in data sets. This is an essential feature of rough sets since real data is often incomplete, inconsistent and ambiguous. Besides, the sensors deployed in a sensor field are inexpensive, often unreliable, and prone to failures.

- Rough set theory can be employed to reduce the dimensionality of data set as a preprocessing step to training a learning system. A rough set based feature or attribute reduction algorithm does not transform the data, and it preserves the underlying data semantics. Its only reliance on simple set operations makes it suitable as a preprocessor for many complex systems. Many real-world systems exhibit non-polynomial complexity with respect to attribute dimensionality. For example, large-scale water treatment plant requires a huge number of attributes to monitor water quality by using sensors. By employing rough set theory, the number of attributes can be reduced and inference speed can be significantly improved.

- Rough set theory can be used to automate or semi-automate a rule-based expert system since it can automatically induce *if-then* rules from empirical data. Automated expert systems can be very useful for real time decision making in several application areas, such as precision agriculture. In precision agriculture, it is possible to automate watering plants in right time with right amount by estimating the soil moisture, humidity, and temperature in arid regions by using sensors.

- Rough sets based algorithms are well suited for parallel processing (Pawlak, Polkowski, & Skowron, 2005). Since the events in sensor networks are often distributed, concurrent, asynchronous, and non-deterministic, a Rough Petri net model can be useful for formal inference.

- Rough sets can deal with both qualitative and quantitative input data. Since rough set based feature reduction technique preserves the underlying semantics of the data, unlike Principle Component Analysis (PCA), RST is recommended as a preprocessing tool for symbolic or descriptive data.

- RST has greater flexibility to capture various aspects of incompleteness or imperfectness in data since it is generalized from classical set theory.

3.3.1.3 The Basic Idea of Rough sets

Real data is often imprecise, incomplete, ambiguous and superfluous. It is important to remove the irrelevant information and derive underlying

knowledge about the data by representing it in the form of rules, equations, or algorithm. Rough set theory provides mathematical tools for reasoning over imprecise and ambiguous data by lowering the degree of precision in data and deriving underlying rules. Rough set theory expresses vagueness by employing a boundary region of a set. If the boundary region of a set is empty, the set is crisp (precise), otherwise the set is rough (imprecise).

3.3.1.4 Rough Sets and Information Tables

In a rough set framework, data is represented by a two-dimensional table (i.e. matrix), called an information system. Rows of the information table are leveled by objects, columns of the table by attributes, and entries of the table are attribute values. Each object is characterized by its condition and decision attribute values. We can define an information system as S in terms of a pair (U, A), where U is a non-empty finite set of objects called the universe and A is a non-empty finite set of attributes, i.e. $S = (U, A)$. Each attribute a $\in A$ can be considered as a function that maps elements of U into a set V_a , where V_a represents the value set of attributes, such that

$$a: U \rightarrow V_a$$

A decision system can be represented as $S = (U, A \cup \{d\})$, where $d \notin A$ is the decision attribute and V_d is assumed to be the set of values of d.

Examples 3.1:

U = {1, 2, 3, …, 14}, A = {a, b, c, e} and decision is represented by d.

Table 3. 1: Temporal information systems

	a	b	c	e	d
1	0	0	1	0	1
2	0	1	0	1	1
3	0	0	0	1	2
4	0	1	0	1	2
5	1	1	1	1	3
6	1	1	0	0	3
7	1	0	1	0	3
8	1	0	0	1	2
9	0	0	1	1	2
10	0	0	1	0	1
11	0	1	0	1	2
12	1	1	0	0	1
13	0	0	1	0	1
14	1	1	1	1	3

3.3.1.5 Indiscernibility or Equivalence Relations

We can define the Cartesian product (U x V) as U x V = {(u, v) | u $\in$ U, v $\in$ V},

where (u, v) represents an ordered pair. A binary relation, R, is a subset of U x V.

If V = U, the Cartesian product becomes U x U and binary relation is a subset of

U x U. We can define various kinds of relations on U depending on the specific

criteria. For instance, R is an equivalent relation, if

- Reflexive, i.e., (u, u) for every u $\in$ U.

- Symmetric, i.e. (u, v) implies (v, u) for every u, v $\in$ U.

- Transitive, i.e., (u, v) and (v, w) imply (u, w) for every u, v, w $\in$ U.

A partition is induced by the equivalence relation R and the subsets generated are called equivalence classes. A partition of a set *U* is a set of nonempty subsets of *U* (e.g. X_1, X_2, X_3,…, X_n} such that $X_1 \cup X_2 \cup … \cup X_n = U$ and $X_i \cap X_j = \varnothing$ for i ≠ j. A partition divides a set into a number of disjoint subsets (or blocks) so that the elements in the same subset are related and elements in different subsets are unrelated. An indiscernibility relation expresses the pair of objects that we cannot discern. The universe (*U*) can be partitioned by the equivalent relations (*R*) and the subsets are called the equivalence classes (Munakata, 2008). We can define equivalence relations and determine the partitions for condition as well as decision attributes.

Example 3.2: Equivalence classes

For a decision table (Table 3.1), we can define the equivalence relation based on the condition attributes (a, b, c, and e) and derive equivalence classes *X1*, *X2*, *X3*, *X4*, *X5*, *X6*, *X7*, and *X8* such that *X1*= {1, 10, 13}, *X2* = {2, 4, 11}, *X3* ={3}, *X4* = {5, 14}, *X5* = {6, 12}, *X6* = {7}, *X7* = {8}, and *X8* = {9}. We can also derive equivalence classes (e.g. *Y1*, *Y2*, and *Y3*) from Table 3.1 based on the decision attribute (d) such as $Y1 \cup Y2 \cup Y3 = U$ and $Y1 \cap Y2 = Y1 \cap Y3 = Y2 \cap Y3 = \varnothing$ where,

Y1 = {1, 2, 10, 12, 13}

Y2 = {3, 4, 8, 9, 11}

Y3 = {5, 6, 7, 14}

It is important to find the mappings from the partitions induced by the condition attributes to the partitions induced by decision attributes.

3.3.1.6 Partition Induced by Condition and Decision Attributes

We can create a partition *R1'* induced by the relation *R1*= {(u,v)|u and v have the same values for a, b, c, e}. Thus, *R1'* = {*X1, X2, X3, X4, X5, X6, X7, X8*}, where *X1*= {1, 10, 13}, *X2* = {2, 4, 11}, *X3* ={3}, *X4* = {5, 14}, *X5* = {6, 12}, *X6* = {7}, *X7* = {8}, *X8* = {9}. We can also create a partition *R2'* induced by the relation *R2* = {(u, v)|u and v have the same values for d}. Thus, *R2'* = {*Y1, Y2, Y3*}, where *Y1* = {1, 2, 10, 12, 13}, *Y2* = {3, 4, 8, 9, 11}, and *Y3* = {5, 6, 7, 14}. These sets in a partition are called concepts (e.g. *Y1, Y2*).

3.3.1.7 Approximation Spaces: Lower and Upper Approximations

In general, the equivalent class in the partition, induced by the above two condition attributes, does not map exactly to a concept in the partition, induced by the decision attributes. Elements in an equivalent class map to different concepts and this arises because of inconsistent information tables, where the same condition attributes do not lead to the same decision attributes. We can define approximation spaces as *S = (U, R)*, where *U* is a finite set of objects and *R* $\subseteq U \times U$ is an equivalent (or indiscernibility) relation on *U*. Indiscernibility relations are the main concept in rough sets or approximate sets. The basic idea

behind rough sets is to construct approximations of sets using R (Munakata, 2008).

The lower approximation consists of those objects that certainly belong to X and the upper approximation consists of the objects that possibly belong to X. The *boundary region* consists of the objects that we cannot decisively determine whether a member or nonmember of X. The *boundary region* is defined as the difference between the upper and the lower approximations. The negative region or outside region consists of the objects that are certainly non-members of X. The outside region is defined as the complement of the upper approximation. For $X \subseteq U$, the definitions are as follows:

Lower approximation ($\underline{S}X$): $\underline{S}X = \{x \in U : [x]_s \subseteq X\}$

Upper approximations ($\overline{S}X$): $\overline{S}X = \{x \in U : [x]_s \cap X \neq \varnothing\}$

Boundary region (BND$_s$(X)): $BND_s(X) = \overline{S}X - \underline{S}X$

Negative region (NEG$_s$(X)): $NEG_s(X) = U - \overline{S}X$

Positive region (POS$_s$(X)): $POS_s(X) = \underline{S}X$

$\underline{S}X$ is the union of all elementary sets of S, where each elementary set is a subset of X. $\overline{S}X$ is the union of all elementary sets of S, where each elementary set contains at least one of the members from X. $\overline{S}X - \underline{S}X$ represents the

boundary region, where the elementary set contains elements that are members of upper approximation region but nonmembers of lower approximation region. $U - \overline{S}X$ shows the negative or outside region, where the elementary set contains elements that are members of the universe but nonmembers of the upper approximation region. Positive region $\underline{S}X$ represents the lower approximation of X. Fig. 3.2 shows the graphical representation of lower and upper approximations of a rough set.
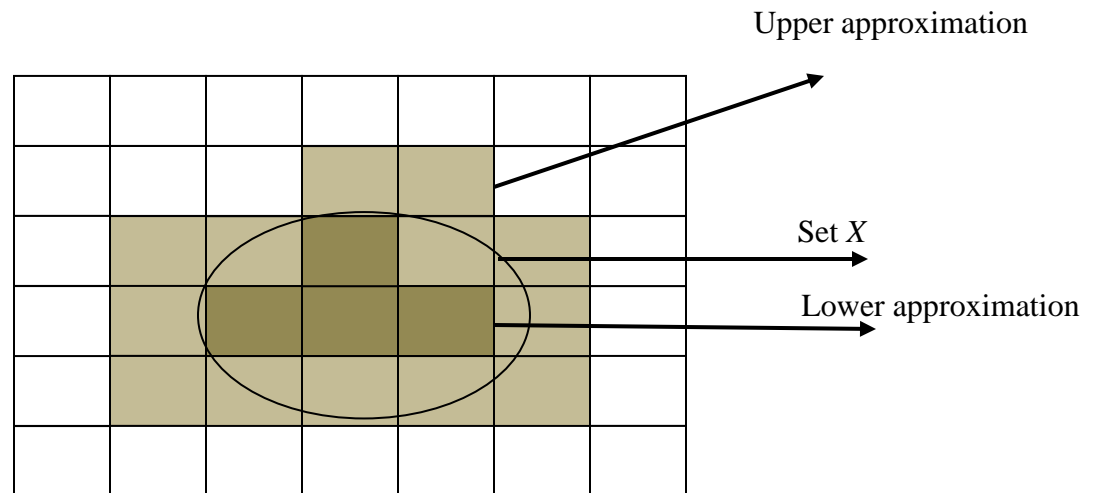
Figure 3. 1: Lower and upper approximations of a rough set

The accuracy of the approximation is measured by $\alpha_s = \dfrac{|\underline{S}(X)|}{|\overline{S}(X)|}$, where $0 \leq \alpha_s \leq$

1. A set is rough if $\alpha_s < 1$ (i.e., X is vague with respect to S). Assuming S and X are equivalence relations in U, the concept of positive region $POS_S(X)$ is defined as:

$$POS_s(X) = \bigcup_{X \in U/D} \underline{S}(X)$$

Example 3.3:

Example 3.3 shows the lower approximation, upper approximation, boundary region and outside region of *X1*, *X2*, *X3* (classification based on decision d) derived from table 3.1.

Equivalent classes in the Universe (*U*):

{1, 10, 13}, {2, 4, 11}, {3}, {5, 14}, {6, 12}, {7}, {8}, {9}

$\underline{S}X1 = \{1,10,13\}$

$\overline{S}X1 = \{\{1,10,13\},\{2,4,11\},\{6,12\}\}$

BND$_s$(X1) = $\overline{S}X1$ - $\underline{S}X1 = \{\{1,10,13\},\{2,4,11\},\{6,12\}\} - \{1,10,13\} = \{\{2,4,11\},\{6,12\}\}$

NEG$_s$(X1) = U - $\overline{S}X1 = \{\{3\},\{9\},\{8\},\{7\},\{5,14\}\}$

$\underline{S}X2 = \{\{3\},\{9\},\{8\}\}$

$\overline{S}X2 = \{\{3\},\{9\},\{2,4,11\},\{8\}\}$

BND$_s$(X2) = $\overline{S}X2$ - $\underline{S}X2 = = \{\{3\},\{9\},\{2,4,11\},\{8\}\} - \{\{3\},\{9\},\{8\}\} = \{\{2,4,11\}\}$

NEG$_s$(X2) = U - $\overline{S}X2 = \{\{1,10,13\},\{7\},\{6,12\},\{5,14\}\}$

$\underline{S}X3 = \{\{7\},\{5,14\}\}$

$\overline{S}X3 = \{\{7\},\{6,12\},\{5,14\}\}$

BND$_s$(X3) = $\overline{S}X3$ - $\underline{S}X3 = \{\{7\},\{6,12\},\{5,14\}\} - \{\{7\},\{5,14\}\} = \{\{6,12\}\}$

$\text{NEG}_s(X3) = U - \overline{S}X3 = \{\{3\},\{1,10,13\},\{9\},\{2,4,11\},\{8\}\}$

The table 3.2 summarizes the equivalence classes in lower approximation, upper approximation, and boundary region for different decision attributes (d).

Table 3. 2: Equivalence classes in different regions in approximation space S.

Equivalent Classes			
	Decision d = 1	Decision d = 2	Decision d =3
Lower Approx.	{1, 10, 13}	{3}, {9}, {8}	{7}, {5, 14}
Upper Approx.	{1, 10, 13} {2, 4, 11} {6, 12}	{3}, {9}, {2, 4, 11}, {8}	{7}, {6, 12}, {5, 14}
Boundary region	{2, 4, 11} {6, 12}	{2, 4, 11}	{6, 12}
Outside region	{3}, {9}, {8}, {7}, {5, 14}	{1, 10, 13}, {7}, {6, 12}, {5, 14}	{3}, {1, 10, 13}, {9}, {2, 4, 11}, {8}

3.3.1.8 Rules Generated from the Decision Table:

Rules can be derived by mapping the partitions induced by the condition attributes to the partitions induced by the decision attributes:

if $X1$ = {1, 10, 13}, then $Y1$ = {1, 2, 10, 12, 13}

if $X2$ = {2, 4, 11}, then $Y1$ = {1, 2, 10, 12, 13}

if *X2* = {2, 4, 11}, then *Y2* = {3, 4, 8, 9, 11}

if *X3* = {3}, then *Y2* = {3, 4, 8, 9, 11}

if X4 = {5, 14}, then Y3 = {5, 6, 7, 14}

if *X5* = {6, 12}, then *Y1* = {1, 2, 10, 12, 13}

if *X5* = {6, 12}, then *Y3* = {5, 6, 7, 14}

if X6 = {7}, then Y3 = {5, 6, 7, 14}

if X7 = {8}, then *Y2* = {3, 4, 8, 9, 11}

Rules can also be defined in terms of attributes:

a(0) AND b(0) AND e(0) => d(1)

a(0) AND b(1) AND e(1) => d(1) OR d(2)

a(0) AND b(0) AND e(1) => d(2)

a(1) AND b(1) AND e(1) => d(3)

a(1) AND b(1) AND e(0) => d(3) OR d(1)

a(1) AND b(0) AND e(0) => d(3)

a(1) AND b(0) AND e(1) => d(2)

Certain rules: The rules generated from the positive region or lower approximations are certain rules. The certain rules for the decision system 3.1 are as follows:

a(0) AND b(0) AND e(0) => d(1)

a(0) AND b(0) AND e(1) => d(2)

a(1) AND b(1) AND e(1) => d(3)

a(1) AND b(0) AND e(0) => d(3)

a(1) AND b(0) AND e(1) => d(2)

Uncertain rules and their confidence factors (α):

The rules induced from the boundary region of the concept are uncertain rules.

For uncertain rules, the confidence factor (α) can be defined as:

$$\alpha = \frac{P|Xi \cap Yj|}{|Xi|}$$

where, Xi and Yj are the equivalence classes based on the condition attributes and decision attributes, respectively. The uncertain rules for the decision system 3.1 are as follows:

a(0) AND b(1) AND e(1) => d(1) OR d(2)

a(1) AND b(1) AND e(0) => d(3) OR d(1)

Confidence factors of uncertain rules can be calculated as follows:

if $X2$ = {2, 4, 11}, then $Y1$ = {1, 2, 10, 12, 13} with α = |{1}| / |{2, 4, 11}| = 1/3

if $X2$ = {2, 4, 11}, then $Y2$ = {3, 4, 8, 9, 11} with α = |{4, 11}| / |{2, 4, 11}| = 2/3

if $X5$ = {6, 12}, then $Y1$ = {1, 2, 10, 12, 13} with α = |{12}| / |{6, 12}| = 1/2

if $X5$ = {6, 12}, then $Y3$ = {5, 6, 7, 14} with α = |{6}| / |{6, 12}| = 1/2

3.3.1.9 Approximation Evaluation

To evaluate the approximations, we can employ some measures, such as sensitivity and specificity of the approximations. Approximation sensitivity is defined as the ratio of the number of objects that can be correctly approximated as members to the actual number of the members. Approximation specificity is defined as the ration of the number of objects that can be correctly approximated as non-members to the actual numbers of the non-members. Approximation accuracy represents the ratio of the total number of correctly approximated objects to the total number of objects. Approximation accuracy can be expressed as a sum of some weighted fractions of sensitivity and specificity (Øhrn, 1999). Table 3.3 represents the performance measures of the approximations for the information table (Table 3.1).

$$sensitivity(S,X) = \frac{|\underline{S}X \cap \text{X}|}{|X|}$$

$$specificity(S,X) = \frac{|(U - \overline{S}X) \cap (U - X)|}{|U - X|}$$

$$accuracy(S,X) = \frac{|X|}{|U|} sensitivity(S,X) + \frac{|U - X|}{|U|} specificity(S,X)$$

Table 3. 3: Evaluation of approximations

	Performance		
	Decision (*d*) = 1	Decision (*d*) = 2	Decision (*d*) = 3
Sensitivity	3 /5 = 0.60	3/5 = 0.60	3/4 = 0.75
Specificity	6/9 = 0.66	8/9 = 0.89	9/10 = 0.90
Accuracy	9/14 = 0.64	11/14 = 0.78	12/14 = 0.86

3.3.1.10 Discernibility Matrix

A discernibility matrix, $M_s(x, y)$, is defined as $M_s(x, y) = \{a \in A \mid \text{discerns } (a, x, y)\}$, where discerns $(a, x, y) \Leftrightarrow a(x) \neq a(y)$. Each entry of the matrix consists of the set of attributes that can be used to discern between objects x and y such that x, y $\in$ U. We need to include only the pairs of distinct objects while constructing the matrix. Since discerns/3 is symmetric and reflexive, $M_s(x, y) = M_s(y, x)$ and $M_s(x, x) = \varnothing$ for all x, y (Øhrn, 1999). Table 3.4 shows the discernibility matrix for the information system table 3.1.

Table 3. 4: Discernibility matrix

	{1, 10, 13}	{2, 4, 11}	{3}	{5, 14}	{6, 12}	{7}	{8}	{9}
{1, 10, 13}	$\varnothing$							
{2, 4, 11}	{b,c,e}	$\varnothing$						
{3}	{c,e}	{b}	$\varnothing$					
{5, 14}	{a,b,e}	{a, c}	{a,b,c}	$\varnothing$				
{6, 12}	{a,b,c}	{a,e}	{a,b,e}	{c,e}	$\varnothing$			
{7}	{a}	{a,b,c,e}	{a,c,e}	λ	{b,c}	$\varnothing$		

{8}	{a,c,e}	{a,b}	λ	{b,c}	{b,e}	{c,e}	∅	
{9}	{e}	{b,c}	λ	{a,b}	{a,b,c,e}	{a,e}	λ	∅

3.3.1.11 Discernibility Function

It is possible to determine the reducts or most informative set of attribute by using discernibility matrix of the information system. The discernibility function f_s is defined as follows:

$$ fs = \prod_{y \in U} \left\{ \sum a \mid a \in Ms(x,y) \text{ and } Ms(x,y) \neq \varnothing \right\} $$

The prime implicants of f_s provide the minimal subsets of attributes.

The discernibility function for the table 3.1 is:

$fs = (b \lor c \lor e) \land (c \lor e) \land (b) \land (a \lor b \lor e) \land (a \lor c) \land (a \lor b \lor c) \land (a \lor b \lor c) \land (a \lor e) \land (a \lor b \lor e)$

$\land (c \lor e) \land (a) \land (a \lor b \lor c \lor e) \land (a \lor c \lor e) \land (b \lor c) \land (a \lor c \lor e) \land (a \lor b) \land (b \lor c) \land (b \lor e) \land (c \lor e)$

$\land (e) \land (b \lor c) \land (a \lor b) \land (a \lor b \lor c \lor e) \land (a \lor e)$

$\equiv (b \lor c \lor e) \land (b) \land (a \lor b \lor e) \land (a \lor c) \land (a \lor b \lor c) \land (a \lor b \lor e) \land (c \lor e) \land (a) \land (a \lor c \lor e) \land (b \lor e)$

$\land (e) \land (b \lor c) \land (a \lor e)$

$\equiv a \land b \land e$

The prime implicant of f_s is $(a \land b \land e)$.

3.3.1.12 Dependency and significance of attributes

Dependency of attributes is one of the important measures to select attributes or features. If the set of decision attributes D totally depends on the set of condition

attributes C, then all values of D are uniquely determined by the attribute set C

and this dependency is denoted by $C \Rightarrow D$. If the set of attribute D partially

depends on the set of attributes C, then some of the values of D are uniquely

determined by the attribute set C and this partial dependency is denoted by

$C \Rightarrow_p D$ where p = $\gamma(C, D)$ and $0 \le p \le 1$. γ(C, D) is defined as:

$$\gamma(C, D) = \frac{|POS_C (D)|}{|U|}$$

where,

$$POS_C(D) = \bigcup_{X \in U/D} \underline{C}(X)$$

$POS_C(D)$ is the positive region of the partition U/D with respect to C. $\gamma(C, D)$

represents the degree of consistency of the decision table or the degree of

dependency between C and D.

Significance of attributes is also another important measure for attribute

reduction. Significance of an attribute represents the importance of the attribute

for decision making. It can be evaluated by observing the impact of its removal

from the information table. Let an attribute $a \in C$ be removed from C, then the

degree of consistency changes from $\gamma(C, D)$ to $\gamma(C - \{a\}, D)$. Then, the

significance of the attribute a is calculated as:

$$\sigma_{(C,D)}(a) = \frac{\gamma(C,D) - \gamma(C - \{a\}, D)}{\gamma(C,D)} = 1 - \frac{\gamma(C - \{a\}, D)}{\gamma(C,D)}$$

where $0 \leq \sigma(a) \leq 1$

If $(\gamma(C,D) = \gamma(C - \{a\}, D))$, then $\sigma_{(C,D)}(a) = 0$.

If $\gamma(C - \{a\}, D) = 0$, then $\sigma_{(C,D)}(a) = 1$.

$\sigma_{(C,D)}(a) = 0$ implies that the removal of the attribute from the condition attribute set C does not have any impact on overall decision making as the degree of consistency of the decision table does not change. Thus the sensor, that measures attribute a, is dispensable since it does not have any significant contribution in decision making. $\sigma_{(C,D)}(a) = 1$ implies that the removal of the attribute a from the condition attribute set C has significant impact on overall decision making and all consistent rules will disappear from the decision table. Thus the sensor, that measures attribute a, is indispensable since it is the most significant sensor in decision making. $0 < \sigma(a) < 1$ represents the range of attribute significance and partial dependency (Mal-Sarkar, Sikder, Yu, & Konangi, 2009).

3.3.1.13 Elimination of Redundant Attributes: Reducts and Core

The minimal set of attributes that can categorize the objects correctly is called a reduct. It represents an attribute subset $B \subseteq A$ of an information table such that

after removal of *A – B* superfluous attributes from an equivalent class it preserves the equivalent relation and consequently the set approximations. In other words, no more attribute can be removed from a reduct without changing the equivalent classes. The reduct of an information system is not unique. For a complex problem, there may be many of these minimal reducts. The set of prime implicants of the discernibility function determines the reducts. The intersection of all reducts is called a core which includes the set of most informative attributes. Core can be defined as Core (*B*) = ∩ Reduct (*B*).

Reducts are very useful in applications where the number of attributes is very high, such as large-scale water treatment plant that requires a huge number of attributes to monitor water quality and perform diagnostic detection of faults. Computing all possible reducts is a non-trivial task; however computing prime implicants is an NP-Hard (Wroblewski, 1995) problem. There are reasonably good heuristics to find sufficient number of reducts in an acceptable amount of time. Heuristic algorithms, such as genetic algorithms (Bazan, Skowron, & Synak, 1994) or dynamic reducts (Lech Polkowski & Andrzej Skowron, 1998) can be used to generate a computationally efficient set of minimal attributes.

After obtaining reducts, a set of *if-then* rules can be generated to create a classifier. Once the reducts have been computed, then deriving the decision rule is a simple task of laying the reducts over the original decision table and

mapping the associated values. Such rules derived from the training set can be used to classify new instances for which the decision classes are unknown. However, it is likely that more than one rule may fire to decide a class for a new object. In that case strategies, such as standard voting, are to be adopted to resolve conflicts among candidate rules that recognize the same object (Greco, Matarazzo, & Slowinski, 2002).

Example 3.4

The information system shown in table 3.1 has four condition attributes, such as *a*, *b*, **c** and *e*, and one decision attribute, *d*. Attribute c is redundant or σ(c) = 0 because the removal of this attribute does not cause any change in the equivalent class structure. For all other attributes, σ ≠ 0 reflects their importance in decision making. The attribute set {*a, b, e*} is the reduct of the information system because the elimination of any of these attributes causes collapse of the equivalent class structure. Reduct and core are the same for this information system since there is only one reduct.

3.3.2 Data Streams Processing and Spatio-temporal Patterns

"A data stream is a real-time, continuous, ordered (implicitly by arrive time or explicitly by timestamp) sequence of items. It is impossible to control the order in which items arrive, nor is it feasible to locally store a stream in its entirety"(Golab & Özsu, 2003). The examples of online data streams are

stocktickers, network measurements and sensor data. With the rapid growth of applications on mining data stream, there is an incremental need to perform association rule mining on data stream (Jiang & Gruenwald, 2006). By finding the temporal association of frequently occurred events, we can unearth causative chains of events which are very useful to find out the root causes of persistent faults (Laxman, Sastry, & Unnikrishnan, 2007).

3.3.2.1 Sensor Data Stream Processing

A sensor network can be modeled as a distributed system of sensor data streams that consist of a sequence of data elements which arrive online. A sensor data stream is a set of timestamped tuples and the order of the sensor data stream is derived from the timestamps. The number of data elements in a stream can be unbounded and larger than the storage capacity of the stream processing system and thus, the query processing systems need to process these elements as they arrive without storing and making multiple passes over it. By monitoring and analyzing sensor data streams immediately as they arrive, one can discover new spatio-temporal patterns that help us better understand the monitored environments. Thus instead of storing the entire data stream, the interesting patterns or templates along with their durations can be stored on the processing systems by employing the concept of time windowing.

A windowing mechanism can be used to limit the amount of data that needs to be stored for query processing. Windows can be defined using absolute (fixed), landmark, or sliding intervals. For absolute intervals, both start and end times are explicitly specified (e.g. August 15, 2008). The landmark interval is defined as an interval when only the start time is explicitly specified (e.g. from August 15, 2008 onwards). The sliding intervals are intervals where, neither start time nor end time is explicitly defined but the duration of the interval is specified (e.g. last 2 hours). Based on the timestamps for each input data stream, the stream elements within a particular time window are selected and considered to be active. Thus, the lifetime of sensor data streams and queries are bounded and they consume resources only when they are active. Lifetimes can be specified in terms of explicit start and end times (absolute window), start time (landmark window), or duration (sliding window), depending on how windows have been defined.

Extracting knowledge from multiple distributed data streams in a sensor network environment is a research challenge that needs to be addressed. Traditional database systems and data processing algorithms for analyzing static data sets are not well suited to handle complex, numerous, and continuous queries over data streams and they need to be reinvestigated for continuous, high-speed, and time-varying data streams in a distributed sensor network. Very few works addressed the problem of processing and analyzing data stream

generated from wireless sensor networks (Cantoni, Lombardi, & Lombardi, 2006; Elnahrawy, 2003; Elson & Estrin, 2004).

The sensors that are geographically near to each other are likely to produce similar data and a sensor that is continually monitoring the same environment will produce streams of values which are correlated in time (Kargupta, 2007). Since a large number of sensors are scattered densely in a sensor network, it is likely that sensor data has strong spatio-temporal correlation. Significant reduction in processing and communication can be achieved by taking into account such correlations and generating spatio-temporal patterns. Such spatio-temporal patterns can also be used to reason over imprecise, incomplete, and missing sensor data.

3.3.2.2 Temporal Information System

The temporal information system in a sensor network can be represented in the form: $\mathcal{A} = (U, S)$, where U is the closed universe that consists of nonempty finite set of observations $x_1, x_2, \ldots, x_n$ at time $t_1, t_2, \ldots, t_n$. and S is a nonempty finite set of sensors $s_1, s_2, \ldots, s_n$. For $\forall\ s \in S$, $s : U \rightarrow V_s$, where V_s is a value of sensor s. A descriptor is represented by the expression $s \in V$, where $s \in S$ and $V \subseteq V_s$. Each column of a temporal information system represents sensors, while rows

represent observations in chronological order. The temporal information system,

shown in table 3.5, describes the behavior of 5 sensors at time $t_1, t_2, \dots , t_{30}$.

Table 3. 5: A temporal information system

Temporal Information System					
5 30	S[0]	S[1]	S[2]	S[3]	S[4]
1	5	3	7	8	5
2	4	6	2	4	5
3	5	3	7	8	1
4	5	3	7	4	5
5	4	6	4	4	5
6	5	3	6	2	5
7	4	6	7	4	5
8	5	3	6	2	3
9	2	5	7	4	5
10	5	3	6	4	5
11	5	3	7	8	2
12	5	3	7	4	5
13	4	6	4	4	5
14	5	3	7	8	5
15	4	6	4	4	5
16	2	5	6	4	5
17	4	6	7	8	4
18	2	5	6	8	4
19	5	3	8	8	4
20	3	6	5	8	7
21	5	3	7	8	7
22	5	3	7	8	9
23	4	6	7	8	9
24	5	3	5	7	6
25	2	5	7	8	3
26	4	6	7	8	5
27	5	3	7	8	3
28	4	7	9	7	2
29	5	3	7	8	6
30	4	7	7	8	6

3.3.2.3 Basic idea of temporal templates

One of the aspects of sensor data mining is the analysis of temporal sensor data that changes in time. During the analysis of temporal data, homogenous patterns can be discovered and the discovered patterns can be expressed in terms of production rules *if … then.* It is possible to simulate both supervised as well as unsupervised learning processes by employing these rules. It is also possible to determine the quality of the discovered knowledge by implementing a classifier.

Temporal templates are homogeneous patterns generated at regular intervals from temporal information systems (Synak, 2001). They can be used to reason over missing, fragmented, and incoherent information gathered from an aggregation point in wireless sensor networks. For a given information system, a generalized template is represented by a set of descriptors as: $\Im = \{(s \in V) : V \subseteq V_s\}$. A signal $x \in U$ matches a generalized template $\Im$, if it matches all descriptors of $\Im$. A template can be precise if it has only one-value descriptors or it can be general if it has multiple descriptors. A temporal template for such a system can be defined as $\Re = (\Im, t_s, t_e)$, where $\Im = \{(s \in V) : s \in B, V \subseteq V_s, B \subseteq A\}$, t_s represents the start time, t_e represents the end time of a temporal template, and $1 \le t_s \le t_e \le n.$

Examples of temporal templates are:

$\Re_1 = (\{(s_1 \in \{v\}), (s_2 \in \{p\})\}, 3, 7)$ and $\Re_2 = (\{(s_3 \in \{d\}), (s_4 \in \{h\})\}, 9, 13)$

It is recommended to search for temporal representatives from the set of all templates by taking the optimal one with respect to some quality measures, and to use them for encoding a sequence of templates (Synak, 2003). Each template is represented as an event and a sequence of templates is considered as a sequence of events. The template representatives are used to replace all the templates in a sequence that are closest to the representative templates, thus the total number of unique templates are reduced to the number of representative templates. In general, closeness can be defined as follows:

$$cl\left(\Re_1, \Re_2\right) = \sum_{a \in A} cl_a\left(\Re_1\left(a\right), \Re_2\left(a\right)\right).$$

Example 3.5

Temporal templates are generated by scanning the temporal information system within a particular time window (W) and shifting the window by a fixed amount (step) in each iteration. The temporal templates derived from the temporal information system (table 3.5) are shown in table 3.6 where quality = 50, window size (W) = 10, and step = 2.

Table 3. 6: Temporal templates

Temporal Templates (Quality threshold = 50)			
	Interval	Code	Template
1	0	0	
2	1	1	s0=5 & s1=3

3	1	2	s3=4 & s4=5
4	5	2	s3=4 & s4=5
5	6	0	
6	9	3	s2=7 & s3=8
7	10	1	s0=5 & s1=3
8	10	3	s2=7 & s3=8

Interval represents the iteration or the number of the time moment in which a template starts. Each template is uniquely identified by its code.

$$Number\ of\ intervals\ = \frac{No.\,of\ observations - Window\ Size}{Step\ Size} + 1$$

For, no. of observation = 30, window size W = 10, step = 2:

No. of intervals or iteration = (30 - 10)/2 +1 = 11

The templates generated from the temporal information system are as follows:

Template#0 (code = 0): 0,

Template#1 (code = 1): s0=5 & s1=3,

Template#2 (code = 2): s3=4 & s4=5, and

Template#3 (code = 3): s2=7 & s3=8.

If several templates appear in the same time interval or iteration, they are marked with the same color. For instance, template#1 and templates#2 both appear in the interval 1, and they are marked by the same color. The color of the

background changes periodically to enhance the visibility of the interval where at least one of the templates is different (changed with time). Interval 1 and interval 5 are marked with different colors to reflect the fact that interval 1 has two templates, such as template#1 and template#2, where as interval 5 has only template#2.

The intervals are not recorded if the templates of a given interval do not change in consecutive intervals. There is no entry for intervals 2-4 in the temporal template table since intervals 2-4 have the same templates as interval 1. We can count the number of consecutive occurrences of a set of templates from the table by subtracting the interval number of its first occurrence from the following interval number. For instance, template#1 and template#2 together last for (5-1) =4 consecutive intervals.

3.3.2.4 Quality of a temporal template

The quality of a temporal template can be defined as a function of width, support, number of descriptors, and precision. The width of a temporal template $\Re = (\Im, t_s, t_e)$ is computed by $(t_e - t_s + 1)$. The support of a temporal template $\Re$ is defined as the number of signals during the interval $[t_s, t_e]$ that matches all of the descriptors from $\Im$. The precision of the template is defined as

$$P_s(\Im) = \sum_{(s \in V) \in \Im} p_s^{\Im}((s \in V)) \quad \text{where} \quad p_s^{\Im}(s \in V) \text{ and is a measure of how much a}$$

descriptor $(s \in V)$ is specific (Synak, 2003).

In our design, quality is defined as a function of support as follows:

$$Quality\ of\ template = \frac{No.\ of\ rows\ that\ match\ with\ the\ template}{Window\ size} \times 100\%$$

For window size = 10 and interval 1:

Quality of the template#1 (s0=5 & s1=3) is (6/10) x 100% = 60%

Quality of the template#2 (s3=4 & s4=5) is (6/10) x 100% = 60%

3.3.2.5 Algorithm to Generate Spatio-temporal Patterns and RS Rules:

The sensors, densely deployed in a sensing field, collect information for constantly evolving entities about the environment, like temperature, pressure, humidity, etc. Often the patterns generated by these sensors' readings are redundant and uninteresting. Sending these redundant data to the sink is neither efficient nor economic. Thus we need a technique that can discover interesting spatio-temporal patterns from the data streams immediately as they arrive and then the data stream will be discarded. Only the interesting patterns are sent to the sink. The advantage of this technique is two folds: firstly, it has the potential to significantly reduce the data communications from the cluster head to the

sink; secondly, it offers a new data stream mining technique that employs the concept of a sliding window. The following algorithm has been developed to generate spatio-temporal templates and the rough set rules from sensor data streams (Mannila, Toivonen, & Verkamo, 1997; Paluch & Rzasa, 2005; Rzasa et al., 2004; Synak, 2001, 2003):

Algorithm:
- a) Preprocess sensors' data by transferring the data to a single, integrated file in a convenient format and filling in the missing values by smoothing averages.
- b) Form a hierarchy of sensors that have spatial and temporal correlations by employing a modified agglomerative single linkage clustering algorithm.
- c) Select only one cluster based on the linkage threshold – only the sensors within that cluster are considered for the current research. Discard the readings from other clusters.
- d) Create a temporal information system $\mathcal{A} = (U, S)$ from the remaining sensors' readings.
- e) Generate spatio-temporal templates by scanning the information system within a particular time window (W) and shifting the window by a fixed amount (s) in each iteration. Spatio-temporal templates are defined as $T = \{(v, ts, te)\}$ where ts = start time =1, te = end time= min $\{|U|, te\}$. In each iteration, find the sensor values that occur at least q times where q = a threshold value (quality). Then find all maximal templates that meet the requirements for the iteration. In the end of each iteration update ts and te as $ts = ts + s$ and $te = min\{|U|, te + s\}$. Repeat until $ts >= |U|$ and obtain a series of spatio-temporal templates for the entire information system.
- f) Create a multi-valued decision system from the series of spatio-temporal patterns where the number of condition attributes is k and decision attribute is 1. The *ith* row of the multi-valued decision system is created by placing the spatio-temporal templates generated in the *ith*, *(i+1)th*, …*(i+k)th* iterations.
- g) Remove all irrelevant patterns and generate reducts, a set of most informative patterns for decisions, by employing rough set theory. Then, generate decision rules from the reduced decision table.

The algorithm generates a number of spatio-temporal templates by shifting a time window of a given size across the temporal information system and scanning the elements within that window. By gradually shifting the time window, temporal templates can be compared with respect to some quality thresholds and it is possible that the previously found template is still the best one in the new window. Thus, we can determine the upper and lower bounds [t_s, t_e] of a template where the template is optimal or close to optimal and meets a certain acceptable quality level. The algorithm may not generate any template for some intervals if there is no strong regularity in sensor readings for those intervals. In a highly dynamic environment, where sensor values are changing very frequently, we may not get any useful template at all.

3.3.2.6 Temporal Templates and Quality Threshold Value

The number of templates generated from a temporal information system changes with the quality threshold. The lower quality templates require fewer matches with the row of the temporal information system. Thus, the number of templates is expected to increase when the template quality is low. On the other hand, the higher quality of templates demands more matches of templates with the row of the temporal information system. Therefore, the number of templates is likely to decrease when the template quality is high. As a consequence, we can achieve more data compression with high quality templates, compared to low quality templates. However, in order to understand the impact of template quality on

data uncertainty we need to characterize uncertainty first and then we can quantify uncertainty for different scenarios. We can also determine the pairwise correlation of uncertainty measures and uncover their tradeoffs in different scenarios.
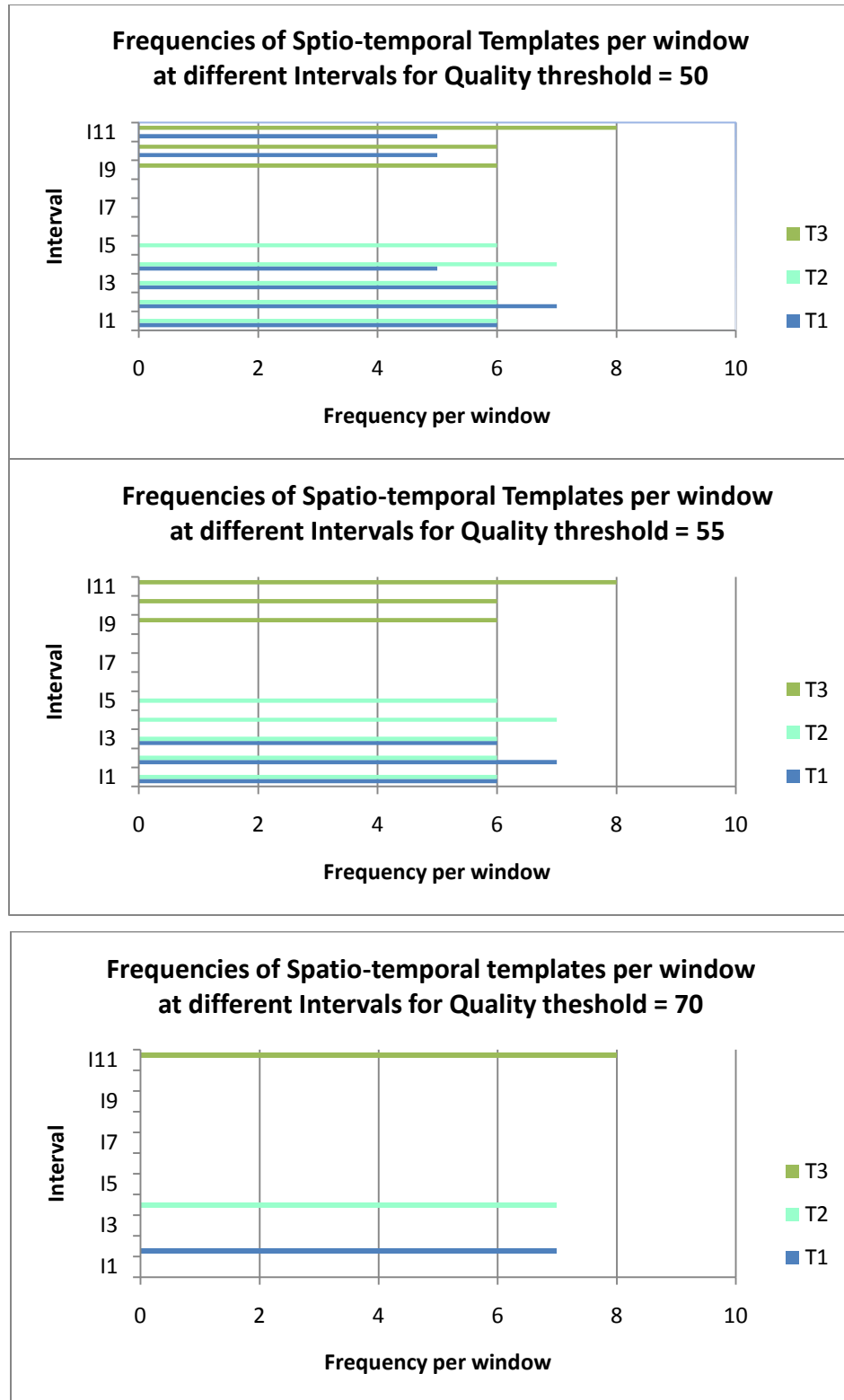
Figure 3. 2: Frequencies of spatio-temporal templates per window for quality threshold a) 50 b) 55, and c) 70 respectively.

Frequencies of temporal templates per window are plotted at different intervals for several quality thresholds: 50, 55, and 70 (Fig. 3.2 a, b, c). Fig 3.2 a shows that for quality threshold = 50, we obtain templates T1 and T2 in intervals 1- 4, only T2 for interval 5, T3 for interval 9, and T1, T3 for intervals 10 and 11. There is no template for intervals 6 – 8. Fig 3.2 b shows that for quality threshold = 55, we obtain T1, T2 for intervals 1-3, T2 for intervals 4 and 5, T3 for intervals 9 - 11. There is no template for intervals 6 – 8. Fig 3.2 c shows that for quality threshold = 70, we obtain T1 in interval 1, T2 in interval 4, and T3 in interval 11. There is no template for intervals 1, 3, and 5 – 10. This demonstrates that the number of temporal templates decreases as we increase the acceptable level of quality threshold.

3.3.2.7 Dependencies between Temporal Templates

We can determine the dependency among temporal templates from a series of temporal templates generated by scanning the temporal information system (Paluch & Rzasa, 2005; Synak, 2001). One set of temporal templates can follow another set of temporal templates. If we know the occurrence of one set of temporal templates, we can predict the occurrence of another temporal template or a set of temporal templates in the future. For example, consider a series of temporal templates: A, B, C, D, E, F, A, C, E, D, F, A, C, E, D, A, F, C. One can determine that the occurrence of C follows the occurrence of A, and the occurrences of D and E in either order is followed by the occurrence of F.

From a series of temporal templates, one can also determine the frequent *episode*, which is a group of events occurring frequently together. An example of frequent episode from the above temporal template series is "the occurrence of C follows the occurrence of F" since the episode occurs several times in this series. There are several data mining and machine learning application areas where these dependencies can be very useful, such as, alarms in a telecommunication network, user interface actions, and occurrences of recurrent illness. By analyzing the on-line alarm stream using these relationships, we can explain the problems that cause alarms and suppress the redundant alarms, and predict severe faults (Laxman et al., 2007; Mannila et al., 1997).

3.3.2.8 Single-valued Temporal Decision System

A series of temporal templates which are disjoint in time are shown in fig 3.3.
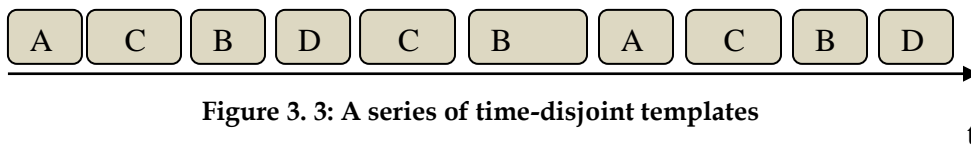
Figure 3. 3: A series of time-disjoint templates

t

A single-valued decision system has a single value for its attributes in a given time moment. In table 3.7, we construct a single-valued temporal decision table from the series of time-disjoint temporal templates (Fig. 3.3) by considering only two consecutive temporal templates (n = 2) from the past as condition attributes and the following temporal template as a decision attribute (Synak, 2001). In

general, n represents the number of templates from the past that are required to

infer the decision template.

Table 3. 7: Single-valued temporal decision table for templates

	$\mathfrak{I}_{t-2}$	$\mathfrak{I}_{t-1}$	$\mathfrak{I}_t$
x_1	A	C	B
x_2	C	B	D
x_3	B	D	C
x_4	D	C	B
x_5	C	B	A
x_6	B	A	C
x_7	A	C	B
x_8	C	B	D

3.3.2.9 Decision Rules from a Single-valued Decision System

We can generate decision rules from the decision table using the rough set

method (Bazan, 1996; Bazan et al., 1994; Pawlak & Skowron, 1993; Synak, 2001).

Table 3. 8: Temporal rules for single-valued decision system.

R1: if $\mathfrak{I}_{t-1}$ = C then $\mathfrak{I}_t$ = B

R2: if $\mathfrak{I}_{t-2}$ = B then $\mathfrak{I}_t$ = C

R3 : if $\mathfrak{I}_{t-2}$ = A then $\mathfrak{I}_t$ = B

R4: if $\mathfrak{I}_{t-2}$ = C and $\mathfrak{I}_{t-1}$ = B then $\mathfrak{I}_t$ = D or $\mathfrak{I}_t$ = A,

                     depending on the width of the template B

R4a: $\mathfrak{I}_t$ = D if width of B=3

R4b: $\mathfrak{I}_t$ = A if width of B=5

The objects x_1, x_4 and x_7 follow the first rule; objects x_3 and x_6 follow the second rule, and x_1 and x_7 follow the third rule (Table 3.8). We have an additional constraint on width of the template for the fourth rule (Table 3.9). The objects x_2, x_8 follow rule R4a and x_5 follows rule R4b.

Table 3. 9: Decision rules depend on the widths of the templates

Objects	Condition Templates		Decision templates
	Width of C	Width of B	
x_2, x_8	4	3	D
x_5	4	5	A

3.3.2.10 Multi-valued Decision System

A multi-valued decision system may have a set of values instead of a single value for its attributes in a given time moment (Paluch & Rzasa, 2005). A multi-valued decision system is constructed from parallel templates (or nondisjoint temporal templates) that occur at the same time interval. The columns represent the attributes and the rows represent the cases or observations arranged chronologically. The last attribute is the decision attribute. Table 3.10 shows the multi-valued decision system generated from the sequence of nondisjoint temporal templates shown in table 3.6. Multiple values for a particular cell are placed in braces and are separated by commas.

Table 3. 10: A Multi-valued decision system for templates

Multi-valued Decision System			
	S[0]	S[1]	Decision
1	{2}	{3}	{1,3}
2	{1,2}	{2}	{3}

3.3.2.11 Temporal Rules from a Multi-valued Decision System

Table 3.11 shows the temporal rules generated from the multi-valued decision system by using the rough set method (Bazan, 1996; Bazan et al., 1994; Paluch & Rzasa, 2005; Pawlak & Skowron, 1993; Rzasa et al., 2004). Each row represents the cases, while each column contains the rules, the "match" for the rules, or the "support" for the rules. The "match" determines the number of cases that match the predecessor of a given rule, while the "support" determines the number of cases that match the predecessor and the successor of a given rule. For the above multi-valued decision system, there is only one temporal rule: (3cs1) => D=1. The rule reflects the fact that if the template#3 occurs at time interval 1, then the next template or decision template will be template#1. "Match" = 1 means that there in only one case where template#3 occurs in interval 1. "Support" = 1 denotes that there is only one case where template#3 occurs in interval 1 and template #1 occurs in the following interval. Thus, for this example the number of temporal rules is 1, the number of consistent rule is 1, and the number of inconsistent rule is 0.

Table 3. 11: Temporal rules for a multi-valued decision system

Temporal rules			
	Rule	Support	Match
1	(3cs1) => D=1	1	1

3.4 Uncertainty Measures of Spatio-temporal Patterns

The uncertainty is a growing research area (Ayyub & Klir, 2006; Klir & Folger, 1988). There are several uncertainty measures that engineers or scientists find useful to quantify different categories of uncertainties. The following uncertainty measures are identified and quantified in the context of spatio-temporal pattern generation in WSN:

- Entropy-based uncertainty that results from conflict among evidential claims.

- Nonspecificity that stems from imprecision associated with the sizes or cardinalities of relevant sets of alternatives.

- Uncertainty based on inconsistent rules derived from the boundary region of concepts.

- Unique template specifies the number of unique patterns in an information system.

- Mean template recurrence is the average frequency of each template in a series of patterns generated from an information system.

- Mean template recurrence variability determines the average variation of a template recurrence from its mean.

- Template vacuity determines the number of uninteresting patterns.

3.4.1 Entropy-based uncertainty

Entropy in information science is a measure of uncertainty or disorder in a message. The more information the message has, the lesser the value of entropy. Entropy in an information system is originated from the concept of entropy in thermodynamics and statistical physics.

The entropy-based uncertainty $H(x)$ in a template generation from temporal information system can be quantified as:

$$H(x) = \sum_{i=1}^{k} p_i \log_2\left(\frac{1}{p_i}\right)$$

where, p_i is the probability of each template i in template space Ω.

Tribus (year) coined the term "surprisal" for $\log_2\left(\frac{1}{p_i}\right)$, which is the degree to which one is surprised to see the result. When the probability is 1, there is no surprise to see the result. As the probability gets smaller and smaller, the surprise goes up and eventually it reaches its maximum, positive infinity. Thus, entropy can be considered as a weighted average of surprisals. If the dataset contains fewer templates, each with higher probability, then it is unlikely that one is surprised very often. On the other hand, the data set with a large number of rare templates is likely to produce frequent surprises.

Examples 3.6

The probabilities of spatio-temporal templates can be calculated from table 3. 6 as follows:

p_1 = P (template#1) = 5/12, p_2 = P (template#2) = 5/12, and p_3= P (template#3) = 2/12 where, $p_1 + p_2 + p_3$ = 1. Once we determine the probability of each template, we can determine the entropy-based uncertainty $H(x)$ for the information system as:

$$H(x) = \sum_{i=1}^{3} p_i \log_2 \left(\frac{1}{p_i}\right)$$

$$= p_1 \log_2 \left(\frac{1}{p_1}\right) + p_2 \log_2 \left(\frac{1}{p_2}\right) + p_3 \log_2 \left(\frac{1}{p_3}\right)$$

$$= 1.48$$

This implies 1.48 bits of uncertainty. Uncertainty depends on the number of templates as well as the split of each template in template space. For instance, if there are a large number of templates but very few templates are favored, then the entropy will be lower compared to the situation when the template space is evenly split. For instance, the entropy of the three equally probable templates can be calculated as 1.58 which is larger compared to the entropy calculated above for the templates generated in table 3.6.

3.4.2 Nonspecificity Measure

This type of uncertainty stems from the lack of specificity resulted from the existence of more than one template at the same time interval and is measured by a well-known Hartley function. When Hartley function $\xi(A)$ is applied to subsets of a given finite template set X, it has the form

$$\xi: P(X) \rightarrow [0,\infty)$$

where $P(X)$ denotes the power set of X, the range of the Hartley function is $0 \leq \xi(A) \leq log_2|X|$, and A is a set of possible templates. The nonspecificity in evidence can be computed by employing the Hartley measure to each subset of templates, and computing a weighted sum of all these measures of the subsets (focal elements) where the basic assignment is used as weight factors. Thus, the nonspecificity $N(m)$ of spatio-temporal templates can be defined as:

$$N(m) = \sum_{i=1}^{n} m(A_i) \log_2(|A_i|)$$

where m is the basic probability assignment for a family of subsets of templates, $A_1, A_2, \ldots, A_n \in P_x$ and m = {m(A$_1$), m(A$_2$), ..., m(A$_n$)}, and m$_i$ = m(A$_i$) for i = 1, 2, ..., n. The $\log_2(|A|)$ represents the nonspecificity of the evidential claim m(A).

When the focal element is singleton, $|X| = 1$, there is no uncertainty. When the focal element is no more singleton (i.e. $|X| > 1$), there are a number of possible templates or decision classes in a proposition. Consider the situation when there is only one focal element X, but the element has three possible templates such as

$X = \{x_1, x_2, x_3\}$. In this case, Shannon entropy based uncertainty is zero since

$p_1 \log_2\left(\frac{1}{p_1}\right) = 0$. However, there is uncertainty regarding the evidence in support

of the decision class. This type of uncertainty stems from the lack of specificity in

evidential claim and is known as nonspecificity.

The Bayesian probability measure fails to estimate the nonspecificity in a body of

evidence. All the focal elements in the probability measure are singleton,

resulting in zero specificity ($|X| = 1$ and $\log_2 |X| = 0$). This shows that

probability measures are inherently fully specific and incapable of characterizing

the nonspecificity dimension of multisource information.

Examples 3.7

The probability assignments of all subsets of spatio-temporal templates can be

calculated from the table 3. 6 as follows:

m_1 = P{1}=0, m_2 = P{2} =1/7, m_3 = P{3} = 1/7, m_4 = {1 ^ 2} = 4/7, m_5 = P{1 ^ 3} =

1/7, m_6 = P{2, 3} = 0, m_7 = P{1 ^ 2 ^ 3} = 0 where the power set (excluding the

null subset) of all templates is P_x ={{1}, {2}, {3}, {1,2}, {1,3}, {2,3}, {1,2,3}} and

$\sum_{i=1}^{7} m_i = 1$. Then, we can determine the nonspecificity-based uncertainty *N(m)*

for the information system (Table 3.6) as:

$$N(m) = \sum_{i=1}^{7} m(A_i) \log_2(|A_i|)$$

$$= m_1 \log_2(|A_1|) + m_2 \log_2(|A_2|) + m_3 \log_2(|A_3|) + m_4 \log_2(|A_4|)$$

$$+ \ m_5 \log_2(|A_5|) + m_6 \log_2(|A_6|) + m_7 \log_2(|A_7|)$$

$$= 0 \times \log_2(|\{1\}|) + 1/7 \times \log_2(|\{2\}|) + 1/7 \times \log_2(|\{3\}|) + 4/7 \times \log_2(|\{1,2\}|)$$

$$+ \ 1/7 \times \log_2(|\{1,3\}|) + 0 \times \log_2(|\{2,3\}|) + 0 \times \log_2(|\{1,2,3\}|)$$

$$= 4/7 + 1/7$$

$$= 5/7$$

3.4.3 Uncertainty from Inconsistent Rules Induced from Boundary Region

When there is inconsistency in the decision table, the equivalence class in the partition, induced by the condition attributes, does not map exactly to a concept in the partition, induced by the decision attributes. Thus, the occurrences of the same spatio-temporal patterns in previous intervals do not lead to the same decision pattern in the following interval. In this situation, an approximation space as $S = (U, R)$ needs to be defined where U is a finite set of templates and $R \subseteq U \times U$ is an equivalence or indiscernibility relation on U and approximation of sets (rough sets) are constructed from R. We are interested in determining to what extent the partition, introduced by the decision attributes, can be characterized or approximated by the partition, introduced by the condition attributes. There are three distinct regions in an approximation space: positive, boundary, and negative regions, explained in section 3.3.1.7. The rules derived from the positive, boundary, and negative regions are consistent, inconsistent (or uncertain), and unidentified rules. By employing the rules from the boundary region of the concept, we cannot certainty predict the causal relationships among

the templates and an uncertainty stems from this inconsistency. This type of uncertainty (U_i), stems from inconsistent rules, for a temporal system can be estimated as:

$$U_i = \frac{I}{R}$$

where I is the number of inconsistent rules derived from the boundary region of concept, and R is the total number rules induced from the both regions, positive as well as boundary regions. This is an important measure for the uncertainty management in pattern generation from sensor data.

3.4.4 Unique Templates

The number of unique templates determines the number of unique spatio-temporal patterns generated from a temporal information system. The number of unique templates is another measure of uncertainty in an information system, and a fewer number of templates is desirable. When the threshold value of template quality is high, the number of unique templates is likely to be low since high threshold requires larger number of observations in the table to match with the templates. When the sliding time window size increases, the number of unique templates is expected to decrease for a particular template quality for the same reason: requirement of larger number of observations to match with the templates. The number of unique templates can be determined by counting the

number of unique template IDs from the series of templates generated from a temporal information table.

3.4.5 Mean Template Recurrence

Mean template recurrence determines the average frequency of the occurrences of a spatio-temporal pattern in a series, generated from a temporal information system. It can be used as a measure of uncertainty and the higher the value of mean template recurrence, the lower the value of uncertainty in information system. Some spatio-temporal patterns are preferred and occur more frequently than others in a series and thus, it is important to measure the mean template recurrence to quantify uncertainty in an information system. Mean template recurrence can be computed as follows:

$$Mean\ Template\ Recurrence\ (\bar{F}) = \frac{\sum_{i=1}^{N} F_i}{N}$$

where F_i is the frequency of the template i and N is the number of total templates generated from the information system.

3.4.6 Mean Template Recurrence Variability

Mean template recurrence variability determines the average variation of the frequency of occurrences of a spatio-temporal pattern from the mean recurrence.

It is a measure of uncertainty and it is not desirable. The higher the value of mean template recurrence variability, the higher the uncertainty associated with spatio-temporal patterns. Once we determine the mean template recurrence, mean template recurrence variability can be computed by measuring its deviation from the mean value and averaging over all templates generated from the information system as follows:

$$Mean\ Template\ Recurrence\ Variability\ (\bar{R}) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(F_i - \bar{F})^2}$$

where F_i is the frequency of template i and $\bar{F}$ is the mean template recurrence.

3.4.7 Template Vacuity

Template vacuity determines the number of vacuous templates in an information table, which implies that there is no interesting pattern in the data. It is a measure of uncertainty and it is not desirable. The higher the template vacuity in the information system, the higher the uncertainty in decision-making since the vacuous templates do not provide useful information. The templates for which the quality is below a certain threshold are considered as vacuous templates or uninteresting patterns and identified as ID = 0. The template vacuity of an information system can be determined by counting the number of templates with

ID = 0. Clearly, template vacuity depends on the sliding window size and the template quality used for template generation. With the increase of window size, more templates cannot meet the quality threshold because the increased number of objects are expected to match with the template. For the same reason, if the template quality is increased up to certain value, template vacuity can increase depending on window size and data type.

3.5 Research Methodology

Fig 3.4 shows the methodological workflow diagram of the current research where a square or a rectangle represents a data object, a rounded rectangle represents an activity, a solid and a dashed directed line denote control flow and data object flow respectively. Functionally, the workflow diagram can be partitioned into three phases:

- Pre-processing phase: The temporal information system (TIS) required for discovering spatio-temporal patterns is created in this phase. However, before creating a TIS, a number of preprocessing steps that includes data cleaning, data sampling, data clustering, data discretization, and data encoding are performed. A clean dataset is obtained by replacing the missing sensors' data by smoothing averages and by converting data into a convenient format. Then the sensors that are spatially and temporality correlated are indentified by employing the clustering algorithm on a clean dataset.

Before we use data for analysis, the size of the attributes value set needs to be reduced to ensure that the number of patterns is not too large and the rules are not too specific (Al-Qaheri et al., 2008). By using the discretization technique, we can determine the cuts in the dataset that divide the continuous range of data into a number of discrete intervals, and a level is assigned for each interval. Thus, a discrete TIS can be obtained by replacing the continuous attribute values in the table with their corresponding discrete values. A numeric TIS is created by using a ceiling function on continuous attribute values. A Boolean TIS is generated by encoding the continuous values as 1 if they are above a threshold value and as 0 if they are below the threshold. The discrete TIS, numeric TIS, and Boolean TIS are useful for symbolic, quantitative, and outlier data mining respectively.

- Spatio-temporal pattern discovery and rule generation phase: the final goal of this phase is to generate rules from TIS and evaluate potential data compression for transforming data into rules. There are several steps that need to be performed to achieve this goal, such as the discovery of spatio-temporal patterns, the creation of a multivalued decision system, the computation of reducts (a set of most informative patterns), the derivation of rules from reducts, the splitting of the dataset, the evaluation and validation of rules.

Once we have the TIS from the previous phase, the spatio-temporal patterns that meet some quality threshold, explained in sections 3.3.2.4 and 3.3.2.5, are discovered by scanning the TIS within a particular time window and shifting the time window (W) by a fixed amount (S) in each iteration. A multivalued decision system is created from a series of spatio-temporal patterns by considering k consecutive patterns from the past as the condition attributes and the following $(k+1)^{th}$ pattern as the decision attribute, as described in sections 3.3.2.6 and 3.3.2.8. A multivalued decision system can have multiple patterns for a particular attribute since it supports parallel patterns in the same interval.

The next important step is to extract and eliminate redundant patterns and determine the most informative patterns for decision making. Redundant patterns are a set of patterns that can be removed from a decision table without changing the degree of dependency between the remaining patterns and the decision and consequently without changing the equivalence relation. The reduct is a minimum set of patterns that preserves the equivalence relation. The reducts can be computed from the discernibility matrix of the decision system by finding the set of prime implicants of the discernibility function, as shown in section 3.3.1.11 (Rzasa et al., 2004).

However, finding all the reducts from a decision system is an NP-Hard problem (Wroblewski, 1995). There are reasonably good heuristic algorithms to find a sufficient number of reducts (Bazan et al., 1994; Lech Polkowski & Andrzej Skowron, 1998). Rough set theory is very useful to determine the reducts by employing two popular attribute reduction measures: degree of dependency (or approximation quality /classification quality) and the information entropy (Al-Qaheri et al., 2008). In this research, the degree of dependency measure is used to compute the reducts.

After obtaining the reducts from the decision table, a simple task of laying the reducts over the original decision table and mapping the associated values is performed. The data compression can be approximated as the ratio of number of rules to the number of objects in the original TIS system since the rule set can be considered as a reduced table where each rule corresponds to one object. Further reduction is achieved by applying the rough set value reduction method.

Then, the set of rules is used as the classifier to validate the rules and to ensure that the rules generated from the information system are self-consistent and closely represent the information system. The dataset is split into two disjoint test datasets. The classifier obtained from the training set is applied on the testing dataset to access the performance. A confusion matrix

is constructed from the training and testing datasets and the accuracy of the methodology of rule induction is estimated. The rule validation is important to ensure that the induced rules faithfully represent the dataset since only the rules, generated at the cluster head, are sent to the sink, instead of the entire datasets. The number of rules is expected to be fewer than the number of observations in datasets because of the use of several data reductions procedures and the spatio-temporal correlations in sensor data; thus this scheme has a potential for data compression. The data compression can be calculated based on the number of rules that are generated at the cluster head (details are provided in chapter 4).

- Uncertainty management phase: This phase includes the characterization and quantification of uncertainty associated with spatio-temporal feature selection, the tradeoff of uncertainty measures for decision making, and hypothesis testing to establish their correlations. Several uncertainty measures are identified and defined in the context of spatio-temporal pattern generation from sensor datasets: entropy, nonspecificity, inconsistent rules, unique templates, mean template recurrence, mean template recurrence variability, and template vacuity. All uncertainty measures are quantified, and their tradeoffs are explained and demonstrated in several graphs. The pairwise correlations of the uncertainty measures are also shown in correlation matrices and the correlation significance is established in a 95%

Figure 3. 4: The methodological workflow diagram
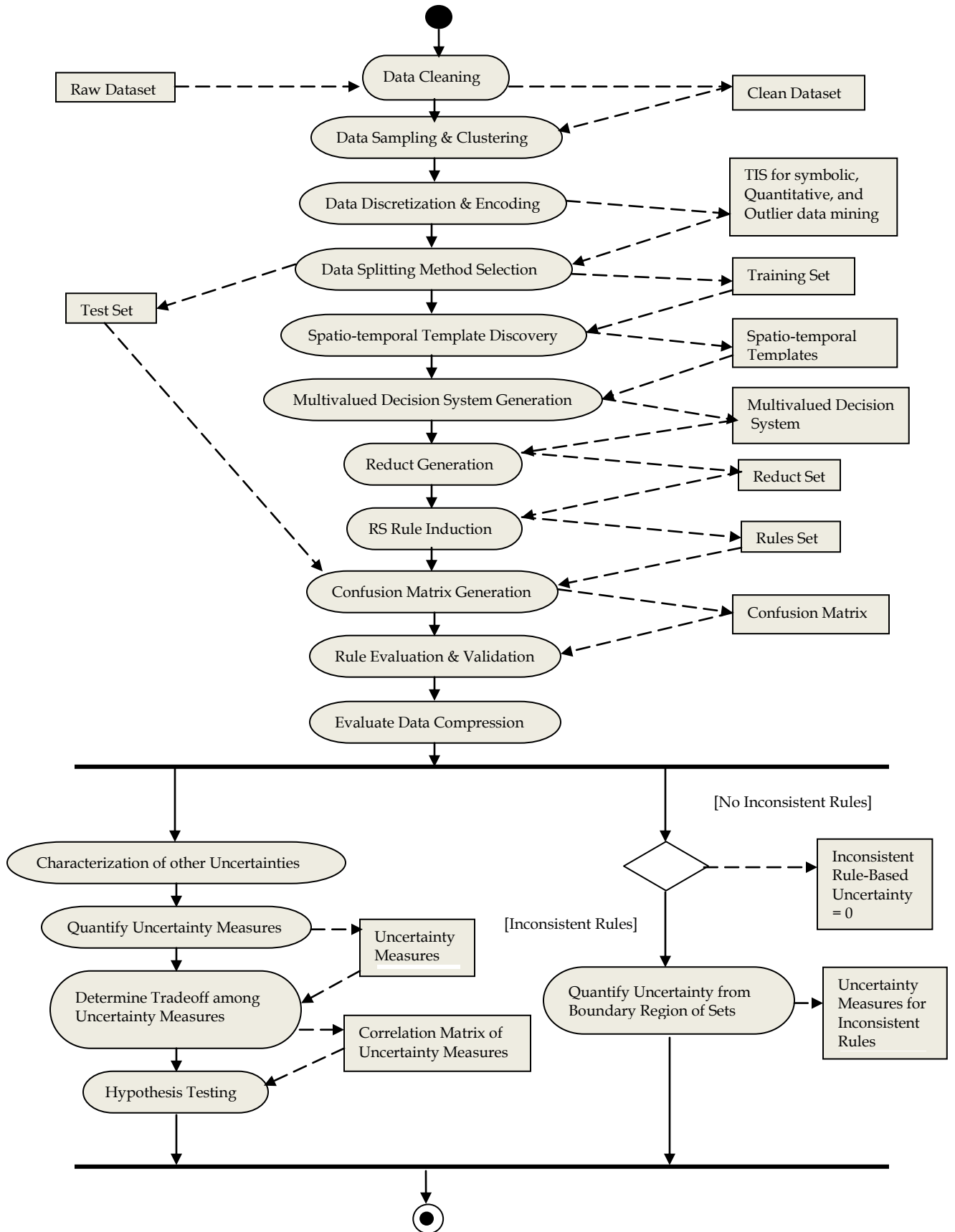
confidence level. With the knowledge of the tradeoffs and the correlations among uncertainty parameters, we can improve our decision making in spatio-temporal feature selection in several application scenarios of stream data processing, including WSNs.

3.6 Rule Validation and Accuracy Estimation

The confusion matrix is an important tool for analyzing the performance of a classifier. It is a table where each column represents the instances in a predicted class, and each row represents the instances in an actual class. The diagonal elements represent correctly specified objects while the off-diagonal elements represent the misclassified objects. The matrix shows the accuracy of the classifier as the percentage of correctly classified objects in a given class divided by the total number of objects in the class that are classified (I. U. Sikder, 2003). Thus, a confusion matrix explains how a classifier behaves for individual classes while the overall accuracy does not indicate that.

Sensor data collected from different sensors are clustered to increase the spatial homogeneity of the data. Then, RS rules are generated from the dataset and the data set is randomly partitioned into two disjoint test datasets to check the rule validity and consistency. To reduce the bias, we randomly select different parts of the data sets as test sets. Finally, the classifier obtained from the training set is

applied on testing data set to access the performance. A confusion matrix is constructed from training and the testing data sets and the accuracy and the consistency of the rules are estimated as follows:

$$Accuracy\ (\alpha) = \frac{\sum_{i=1}^{N} X_{ii}}{N}$$

where $\sum_{i=1}^{N} X_{ii}$ is the number of all diagonal templates that are classified correctly and N is the number of all templates that are classified.

3.7 Correlation Matrix and Statistical Significance

The correlation coefficient matrix (R) is calculated from the input matrix (X) whose rows are observations and whose columns are uncertainty measures. Each element of correlation coefficient matrix (R) is calculated as follows:

$$R(i,j) = \frac{C(i,j)}{\sqrt{C(i,i)C(j,j)}}$$

where the covariance matrix C = cov(X), i represents the row and j represents the column of matrix R.

A matrix (P) can be calculated where each p-value represents the probability of getting a correlation by random chance, but the actual correlation is zero. The p-values are used to test the null hypothesis that there is no correlation among the uncertainty measures. For a pair of uncertainty measures, if the p-value < 0.05 and the correlation coefficient r > 0.5 then the pairs are positively correlated in a 95% confidence level. A pair with p < 0.05 and r < 0.5 implies that they are negatively correlated in a 95% confidence level. The correlation between a pair with p-value > 0.05 is not statistically significant. The confidence bounds are computed based on assumptions that the sample size is large and X has a multivariate normal distribution. Even when the assumptions do not hold, we can use this technique to determine the statistical significance of correlation coefficients by employing bootstrap sampling and generating a large sample when X has a multivariate normal distribution.

By determining whether the pair of uncertainty measures are positively, negatively, or uncorrelated in a 95% confidence level, we can uncover their tradeoffs. Once we know their tradeoffs, we can develop the optimization model based on these parameters, and alleviate decision making even in the presence of uncertainties.

CHAPTER IV
TEMPORAL TEMPLATES GENERATION AND UNCERTAINTY MANAGEMENT

4.1 Chapter Introduction

The objective of this chapter is to provide a validation of a mathematical formalism for the uncertainty management in wireless sensor networks and to provide the validation for the rules generated by a rough set based spatio-temporal pattern discovery scheme. The mathematical foundation of the hybrid model based on rough set and pattern-based data aggregation formalism is established in the previous chapter. The theory of spatio-temporal templates generation is discussed in Chapter 3. In this chapter, we employ the formalism in real world sensor data for finding the homogeneous patterns in sea surface temperature (SST) and generating the association rules. The rules are validated by constructing a confusion matrix from several parts of the patterns. The uncertainties in the pattern generation of sea surface temperature stemming from

156

the imprecise data or missing data from sensors are characterized and quantified. Finally, the correlations among the uncertainty measures are identified and the statistical significance of their correlations is established. This chapter has four major sections: data collection and representation, data preprocessing, spatio-temporal pattern discovery and rule generation, and uncertainty management.

4.2 Data Collection and Representation

Real-time sensor data from moored ocean buoys along the equator in Pacific Ocean, maintained by the Tropical Atmosphere Ocean /Triangle Trans-Ocean Buoy Network (TAO/TRITON) joint project, have been used for this research. Since we are not aware of any publicly available data generated by wireless sensor networks, we attempt to simulate sensor network data from real-time sensor data by using some preprocessing techniques and forming spatial clusters from the time series generated by the TAO/TRITON data. Clustering increases data correlation and homogeneity which are important features of sensor networks. In the dissertation, we attempt to find out the spatio-temporal patterns by exploiting this correlation in data. Thus, we believe that our simulated sensor data will suffice our research requirements.

4.2.1 NOAA and TAO/TRITON Project

National Oceanic and Atmospheric Administration (NOAA) maintains a network of buoys, tidal stations and satellite measurements that provide a

continuous report of the state of the ocean and Great Lakes. NOAA's Tropical

Moored Buoy (TMB) projects consist of the Tropical Atmosphere Ocean

(TAO)/Triangle Trans-Ocean Buoy Network (TRITON), *Pilot Research Moored*

Array in the Tropical Atlantic (PIRATA), and Research Moored Array for African-

Asian-Australian Monsoon Analysis (RAMA). The purpose of the project is to

perform scientific research on warm water in the equatorial ocean and determine

its effect on world climate change. TAO/TRITON buoys are deployed along the

equator in Pacific Ocean; PIRATA buoys are in Atlantic Ocean; and RAMA

buoys are in Indian Ocean (Fig 4.1).
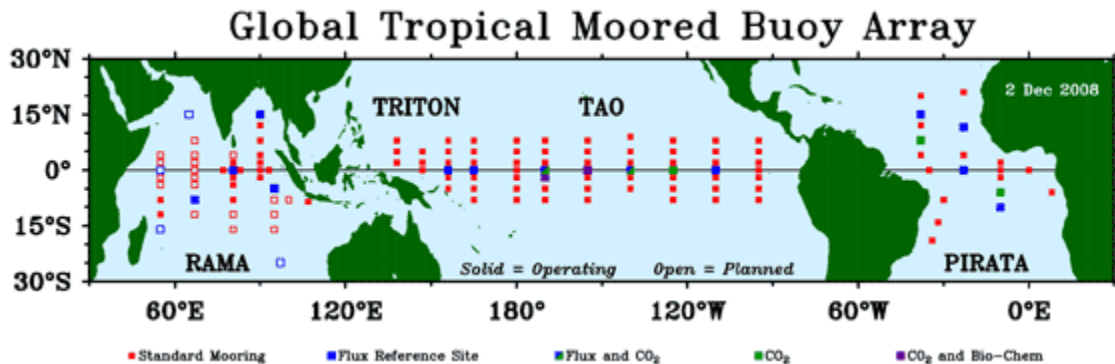
Figure 4. 1: Global Tropical Moored Buoy Array

The TAO/TRITON project has been built over the past 15 years; through the

efforts of many nations, now the TAO project is mainly supported by the

National Oceanic and Atmospheric Administration (United States), and the

TRITON project is supervised by the Japan Agency for Marine-earth Science and

TEChnology (Japan). The TAO/TRITON array consists of approximately 70

moorings in the Tropical Pacific Ocean to store real-time oceanic and atmospheric data via the Argo satellite system. The TAO/TRITON buoys are deployed in the equatorial region of the Pacific Ocean by specially equipped ships, and the buoys are anchored to the ocean floor in water at different depths.

The TAO/TRITON buoys are equipped with sensors that collect oceanic and atmospheric data such as wind speed, wind direction, air temperature, relative humidity, barometric pressure, sea surface temperature, salinity, water pressure, and ocean current. Data collected by the sensors are transmitted to the ground stations and then to the research stations and WWW several times a day through NOAA's polar satellite systems. TAO/TRITON data can be downloaded from the web or via anonymous FTP. The variations of environmental conditions in the tropical Pacific Ocean are illustrated by graphical displays and animations of the data in several formats. TAO/TRITON data are freely available to research community, operational forecasting community, and the general public.

4.2.2 Sea Surface Temperature Data Collection

Our mathematical formalism has been tested on sea surface temperature data from the TAO/TRITON array in the equatorial region of the Pacific Ocean. SST data were downloaded from their web site on an-hourly basis for six months, starting from February 20th, 2005, to August 20th, 2005. Before we downloaded data for this period, we ensured the quality and continuity of data so that our

spatio-temporal pattern-based data aggregation was accurate. Fig 4.2, taken from the TAO/TRITON web site, shows how to display and deliver data with an easy-to-use user interface. The small red solid squares within a rectangle represent the sensors that have been selected to deliver SST data. The temporal resolution of the recoded data can be changed to daily basis or high resolution (once in 10 minutes).
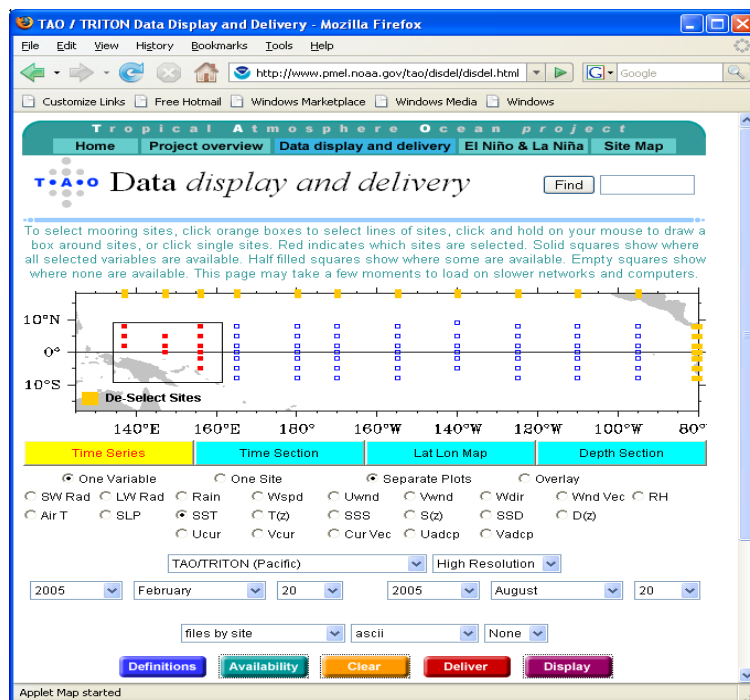
Figure 4. 2: Data display and delivery

4.2.3 Data Availability

Sometimes buoys are broken, disconnected from the anchors, and drift due to tides or strong sea winds, and then the sensors are unable to report measurements. Data is not recorded to their databases for that period of time unless the buoy is replaced or repaired. In general, almost 15 percent of the data

is always missing in the records. This can be determined by inspection of the mooring location time series. To obtain good quality data, we needed to locate where the data availability was high. We found that a high quality sea surface temperature data series was available during the period from February 20, 2005, to August 20, 2005; this data series had high resolution (on an hourly basis). Fig 4.3 shows the available sites for sea surface temperature data at different latitudes and longitudes. We can find if there is a discontinuity in the sensor reading for some period by a simple click on the site.
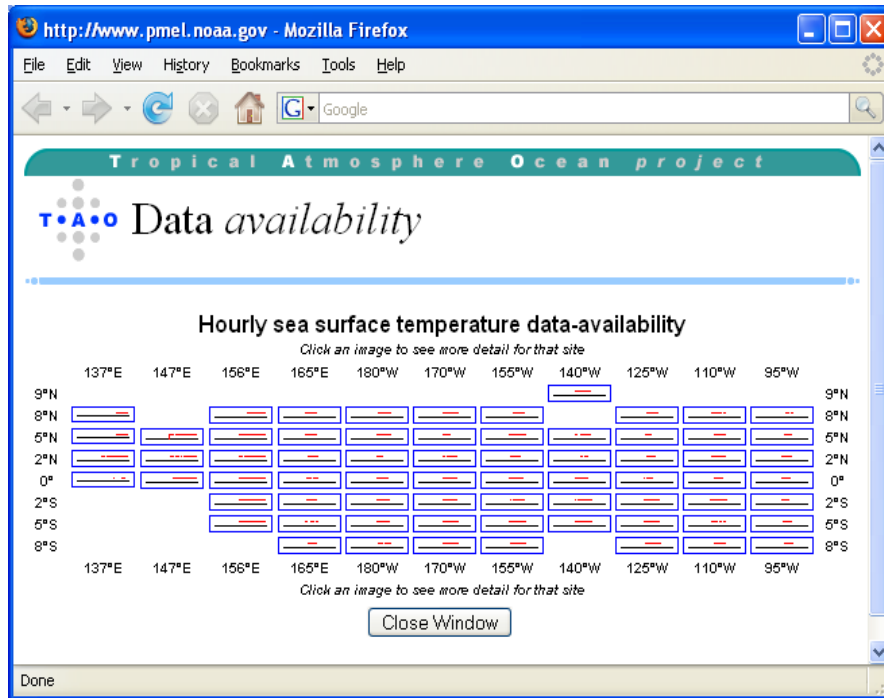
Figure 4. 3: Data Availability

4.2.4 SST Data Format

SST data format includes sensor id, sensor latitude, sensor longitude, the date and time of measuring SST, and the SST values (Table 4.1).

Table 4. 1: Sea Surface Temperatures time series at different locations along the Pacific Ocean.

Sensor ID	Sensor Latitude	Sensor Longitude	Date (YYYYMMDD)	Time (HHMM)	SST (^{0}C)
sst0n147e_hr	0.0000	147.0000	20050220	0000	29.31
sst0n147e_hr	0.0000	147.0000	20050220	0300	29.48
sst2n137e_hr	2.0000	137.0000	20050220	1400	29.05
…	…	…	…	…	…

4.2.5 Time Series Plot

The time series were plotted from the SST data recorded by the sensors implemented on the buoys deployed in the eastern part of the Pacific Ocean during the period from February 20, 2005, to August 20, 2005 (Fig. 4.4).
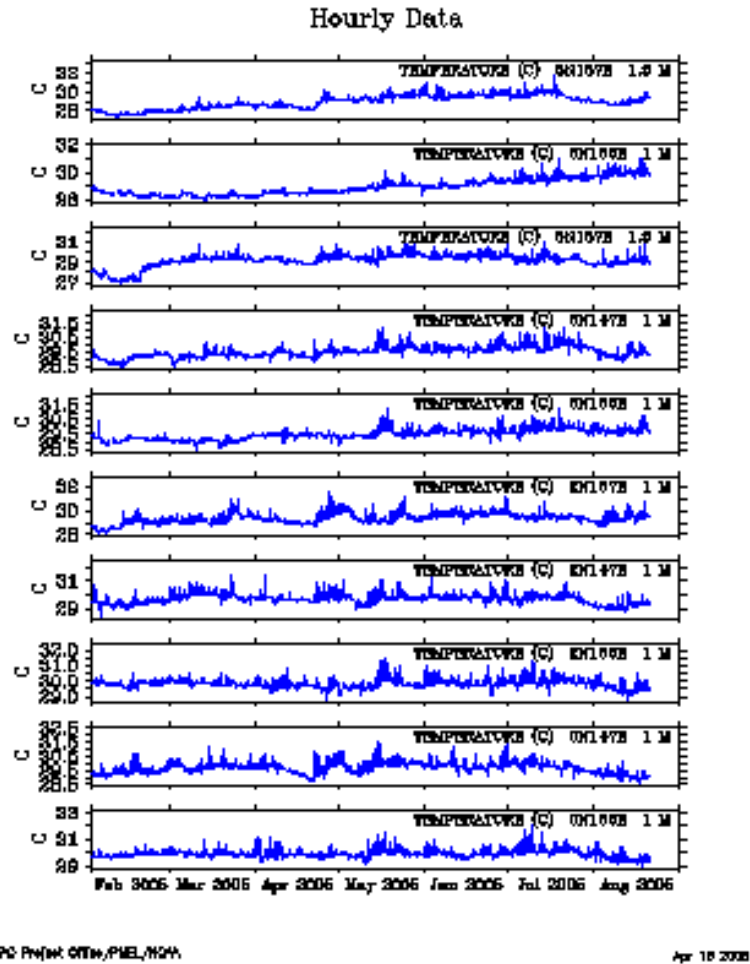
Figure 4. 4: Sea surface temperature data time series

The sensors reported SST every hour for six months, resulting in 4368 observations for each sensor. The location of each sensor was also noted during that period. The time series of SST data clearly show high temporal variations in the data. The spatio-temporal patterns can be discovered from the time series by using the recent history of the readings. We can employ these patterns as templates to predict the most likely future values. We expected a useful number of templates since the time series of SST for the given interval shows significant temporal variations.

4.3 Data Preprocessing

Often data directly collected from the source is raw data, and such data should be preprocessed and converted into an appropriate format before it can be processed by the data mining algorithm. Data preprocessing is one of the most important as well as time consuming tasks in data mining, especially for large data sets. Data sets can be large when the dimensionality is high (e.g. gene expression data) or/and the number of instances is high (e.g. image data). In general, high dimensionality data takes more preprocessing time than the data set with a high number of instances. Data preprocessing involves data cleaning, data integration, data transformation, data reduction and data discretization.

Real world data is often dirty: incomplete (e.g. lacking attribute values, lacking certain attributes of interest, or containing only aggregate values), noisy (e.g. containing errors or outliers), and inconsistent (e.g. containing discrepancies in codes or names). Data should be cleaned to improve its quality before we use it for data mining. There are several methods to clean raw data, such as the binning method, clustering, or regression (Sushmita Mitra & Acharya, 2003). In our data set of SST, approximately 15% data are missing, and these missing data have been replaced by smoothing averages. Missing values also can be replaced by a global constant or the most probable value based on Bayesian formula or decision tree (P. Liu, El-Darzi, & Lei, 2008).

After cleaning, the data have been integrated to a single database from multiple sensors. In the data transformation step, a scaled spatial distance matrix has been obtained by employing min-max normalization, and a normalized feature distance matrix has been created by employing z-score standardization. A weighted matrix is obtained by associating different weights to these two normalized matrices and combining them. By employing a clustering algorithm on the weighted matrix, we can identify the sensors that are spatially close and produce similar SST values and generate a temporal information system (TIS) for a particular cluster. The goal of clustering is to obtain reduced representation in volume that produces the same or similar analytical results by finding the natural groupings in the data.

The last step of the data preprocessing is data discretization and data encoding. It is another data reduction technique but with special importance. Data discretization converts continuous data attribute values into a finite set of intervals with minimal loss of information. Discretization significantly improves the quality of the discovered knowledge and expedites several data mining tasks such as association rule discovery, classification, and prediction. We have discretized the SST time series by using a standard deviation classifier (Table 4.9).

4.3.1 Sensors' Locations

Sensors' locations have been represented by their latitudes and longitudes. However, since lat-long is not a very convenient way of expressing distance, we convert it to a degree decimal notation. Table 4.2 shows the latitude and longitude of the sensors used for our research in degree decimal notation.

Table 4. 2: The Latitudes and longitudes of the sensors

Sensor ID	Sensor Latitude	Sensor Longitude
sst0n147e_hr	0.0000	147.0000
sst0n156e_hr	0.0000	156.0000
sst2n137e_hr	2.0000	137.0000
sst2n147e_hr	2.0000	147.0000
sst2n156e_hr	2.0000	156.0000
sst2s156e_hr	-2.0000	156.0000
sst5n137e_hr	5.0000	137.0000
sst5n147e_hr	5.0000	147.0000
sst5n156e_hr	5.0000	156.0000
sst5s156e_hr	-5.0000	156.0000
sst8n137e_hr	8.0000	137.0000
sst8n156e_hr	8.0000	156.0000

4.3.2 Conversion of Sensors' Spatial Distance to Great Circle Distance

The distance between two sensors is not measured in Euclidean distance; it is measured in great circle distance (non-Euclidean). A great circle is a section of a sphere that contains a diameter of the sphere. The great circle distance is the shortest distance between any two points on the surface of a sphere (Fig 4.5).

Small circles
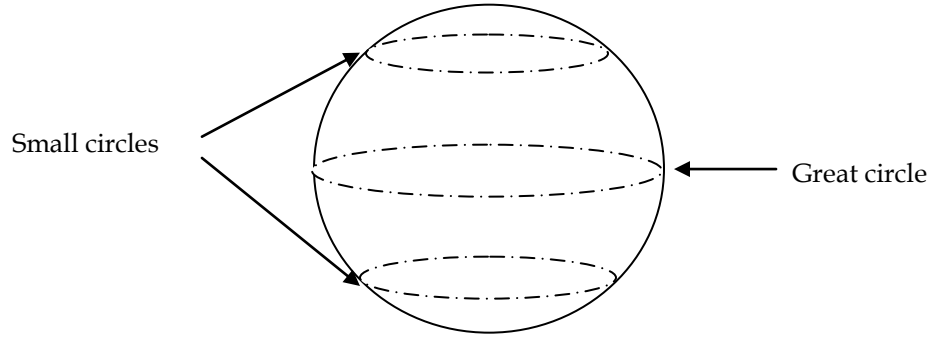
Great circle

Figure 4. 5: Great circle of a sphere

Great-circle distance (*d*) for radius r is given by: $d = r\Delta\hat{\sigma}$

where,

$$\Delta\hat{\sigma} = \arctan\left(\frac{\sqrt{(\cos\phi_f \sin\Delta\lambda)^2 + (\cos\phi_s \sin\phi_f - \sin\phi_s \cos\phi_f \cos\Delta\lambda)^2}}{\sin\phi_s \sin\phi_f + \cos\phi_s \cos\phi_f \cos\Delta\lambda}\right)$$

(φ_s, λ_s); (φ_f, λ_f) are the geographical latitude and longitude of two points respectively.

$\Delta\varphi$, $\Delta\lambda$ are their differences, and $\Delta\hat{\sigma}$ is the spherical distance or angular difference.

4.3.3 The Spatial Distance Matrix (in Meter)

The distance between two sensors is calculated based on geodesic curves, using geographic coordinates projected on to the 'GRS 80' Spheroid. The spatial distance matrix is shown in Table 4.3.

Table 4. 3: Spatial distance matrix

Sensor ID	sst0n147e_hr	sst0n156e_hr	sst2n137e_hr	sst2n147e_hr	...
sst0n147e_hr	0	1007908.5	1134728.6	221149.45	...

sst0n156e_hr	1007908.5	0	2117206.4	1007606.7	…
sst2n137e_hr	1134728.6	2117206.4	0	1112519.6	…
sst2n147e_hr	221149.45	1007606.7	1112519.6	0	…
…	…	…	…	…	…

4.3.4 The Normalized Spatial Distance Matrix

The distance between sensors is transformed by using max-min normalization technique so that the distance value is scaled to fall in the range of $0 \leq S_v \leq 1$.

$$S_v = \frac{S_c - Min}{Max - Min}$$

Table 4.4 shows the normalized spatial distance matrix.

Table 4. 4: The normalized spatial distance matrix

Sensor_ID	sst0n147e_hr	sst0n156ehr_	sst2n137e_hr	sst2n147e_hr	…
sst0n147e_hr	0	0.3673942	0.419318	0.0452723	…
sst0n156e_hr	0.367394195	0	0.8215727	0.3672706	…
sst2n137e_hr	0.419317976	0.8215727	0	0.410225	…
sst2n147e_hr	0.045272342	0.3672706	0.410225	0	…
…	…	…	…	…	…

4.3.5 The Feature Distance Matrix

The feature distance is transformed by employing the z-score standardization technique where a z-score is calculated as follows:

$$Z = \frac{X - \bar{X}}{\sigma}$$

$\bar{X}$ represents the average feature distance and σ specifies the standard deviation.

The feature distance matrix is shown in Table 4.5.

Table 4. 5: The feature distance matrix

	sst0n147e_hr	sst0n156e_hr	sst2n137e_hr	sst2n147e_hr	...
sst0n147e_hr	0	87.38309106	89.86426229	57.77247292	…
sst0n156e_hr	87.38309106	0	92.93030761	86.43789524	…
sst2n137e_hr	89.86426229	92.93030761	0	89.16191319	…
sst2n147e_hr	57.77247292	86.43789524	89.16191319	0	…
...	…	…	…	…	…

4.3.6 The Scaled Feature Distance Matrix

The scaled feature distance matrix is obtained from the previous matrix by employing the max-min normalization scheme on the data (Table 4.6).

Table 4. 6: The scaled feature distance matrix.

	sst0n147e_hr	sst0n156e_hr	sst2n137e_hr	sst2n147e_hr	...
sst0n147e_hr	0	0.570634163	0.618449479	0	…
sst0n156e_hr	0.570634163	0	0.67753606	0.55241904	…
sst2n137e_hr	0.618449479	0.67753606	0	0.604914321	…
sst2n147e_hr	0	0.55241904	0.604914321	0	…
...	…	…	…	…	…

4.3.7 The Matrix by Combining Normalized Spatial and Feature Distances

The weighted matrix is obtained by associating different weights to the normalized spatial and feature distance matrices and then combing the two matrices (Table 4.7).

In general, weighted matrix = $S = S1^* \omega_1 + S2^* \omega_2$; for research model, $S = S1^*0.3+S2^*0.7$

Table 4. 7: The weighted matrix by combining normalized spatial and feature distances

	sst0n147e_hr	sst0n156e_hr	sst2n137e_hr	sst2n147e_hr	...
sst0n147e_hr	0	0.509662	0.55871	0.013582	…
sst0n156e_hr	0.509662	0	0.720747	0.496875	…
sst2n137e_hr	0.55871	0.720747	0	0.546508	…
sst2n147e_hr	0.013582	0.496875	0.546508	0	…
...	…	…	…	…	…

4.3.8 The Clustering of Sensor Data

Clustering is a useful technique for discovering knowledge from a sensor dataset. It is considered an unsupervised learning when data label is undefined. Clusters are natural groupings of sensors based on the similarities between them. The distance between sensors in a cluster is less than the distance between a sensor in a cluster and any sensor outside the cluster. By clustering sensor data,

we can find out the sensors in the TAO/TRITON network that are proximal and have similar SST measurements. Once the sensors are identified, we can generate temporal information system (TIS). The advantages of clustering in WSN that come from in-network data aggregation are as follows:

- Increases data homogeneity

- Eliminates data redundancy

- Reduces global communication

- Improves resource efficiency

- Prolongs network lifetime

- Provides load balancing

- Removes outliers

4.3.8.1 The Clustering Algorithms for WSN

The clustering algorithms for WSN are discussed in detail in Chapter 2. However we cannot use them for the current research since most of them require at least one real time parameter, which is not available in the public domain of the TAO/TRITON project. For instance, the energy-based algorithms LEACH and HEED require information about the energy levels of the sensors in real time. The Weighted Clustering Algorithm (WCA) elects a node as a cluster head based on the number of neighbors, transmission power, battery life and mobility rate of the node (Chatterjee, and, & Turgut, 2002; Chatterjee, Das, & Turgut, 2002). Hence we selected a simple and basic clustering algorithm in data mining,

namely, the hierarchical agglomerative clustering algorithm, with some modification, to meet our research requirements. It is noted that the main focus of our research is not clustering but management of uncertainty stemming from spatio-temporal patterns in WSN. We use clustering as a preprocessing step of data preparation to simulate WSN data and to increase data homogeneity.

Hierarchical clustering algorithms create hierarchical nested partitions of the dataset by using a tree-structure called a dendogram and some termination criteria. It can be categorized as agglomerative or divisive. Hierarchical agglomerative clusters are formed in a bottom-up fashion, starting from the individual sensors at the leaves as separate clusters and iteratively progressing upward by merging closest clusters until all sensors belong to one cluster. On the other hand, divisive algorithm clusters are created in a top-down fashion, starting from a single cluster at the root and iteratively progressing down to the leaves by splitting into clusters. The merging of clusters can be achieved mainly in two different ways: single linkage and complete linkage. In single linkage, two clusters will combine if the minimum distance between two sensors from two different clusters is the least. On the other hand, complete linkage combines two clusters if all sensors in one cluster are close to all sensors in the other (Sushmita Mitra & Acharya, 2003). Our algorithm is based on single linkage agglomerative clustering.

4.3.8.2 The Modified Graph-theoretic Single Linkage Clustering Algorithm

The algorithm focuses on the weighted sum of the spatial distance and feature distance. This is an extension of agglomerative hierarchical algorithm. The nodes represent sensors and the edges denote the proximities between sensors. The proximity matrix (S) is given by (S) = w1 * S1 + w2 *S2 where, S1 is a scaled spatial distance matrix and S2 is a scaled attribute distance matrix; w1 and w2 are weights associated with the scaled spatial distance matrix and attribute distance matrix, respectively. They are generic and can be defined based on the applications.

Single linkage clustering algorithm:

Input: proximity matrix; Output: sets of clusters.

Let be $D(x, y)$ the distance between clusters x and y and $N(x)$ the nearest neighbor of cluster x.

1. Initialize as many clusters as the number of sensors
2. For each pair of clusters (x, y) compute $D(x, y)$
3. For each cluster x compute $N(x)$
4. Repeat until obtain the desired number of clusters
   a) Determine x, y such that $D(x, y)$ is minimized (i.e $D(X, Y) = \min_{x \in X \, y \in Y} d(x, y)$)
   b) Agglomerate cluster x and y
   c) Update each $D(x, y)$ and $N(x)$ as necessary
5. End of repeat

The cluster function can be used to prune branches off the bottom of the tree and assign all the objects below each cut to a single cluster and creates a partition of

the data. The cluster function can create these clusters by detecting natural groupings in the hierarchical tree or by cutting off the tree at an arbitrary point.

4.3.8.3 The Dendrogram for Clustering Spatially Close Sensors
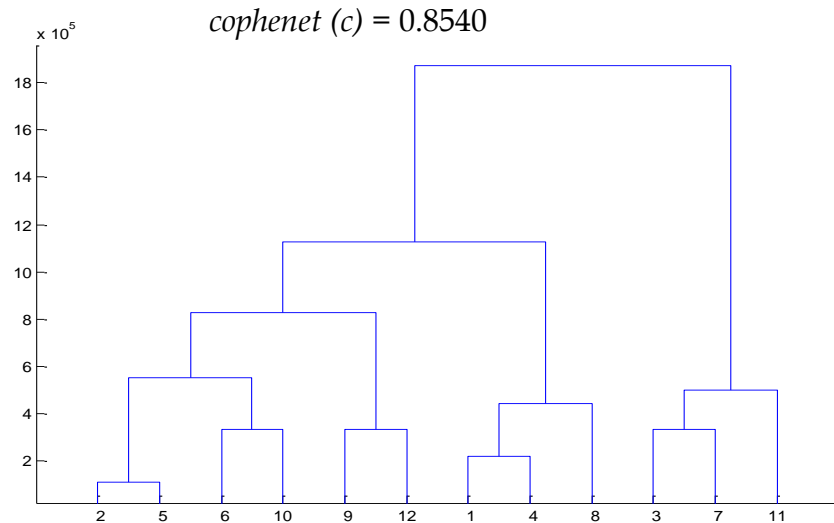
Figure 4. 6: A dendrogram for clustering sensors which are spatially close

Leaves represent sensors and the length of the paths between leaves represents the spatial distance between sensors. Initially each cluster forms a singleton cluster so that there are N singleton clusters for N sensor nodes. Then the two closest sensors merge into a single cluster. This process continues until all sensors belong to a single cluster. In Fig 4.6, sensors 2 and 5 merge into a single cluster since they are the closest neighbors, then sensors 1 and 4 combine, and so on. In each iteration, the minimum distance between the clusters is recalculated as cluster formation continues and the decision should be made based on the

updated distance. The details of node linkage at different threshold distances are

shown in Table 4.8.

4.3.8.4 The Sensor Node Linkage at Different Thresholds

The leaf nodes in the cluster hierarchy are the sensors in the original dataset,

numbered from 1 to m. They are the singleton clusters from which all higher

clusters are built. Each newly formed cluster, corresponding to row i in z, is

assigned the index m+i, where m is the total number of initial leaves (sensors).

Table 4. 8: The node linkage at different threshold

Node	Node	Threshold Distance
2	5	0
1	4	0.045272
6	10	0.09055
3	7	0.09055
9	12	0.090562
8	14	0.135822
11	16	0.158473
13	15	0.181092
17	20	0.294286
18	21	0.415376
19	22	0.721141

4.3.8.5 The Iterations of Clustering Algorithm
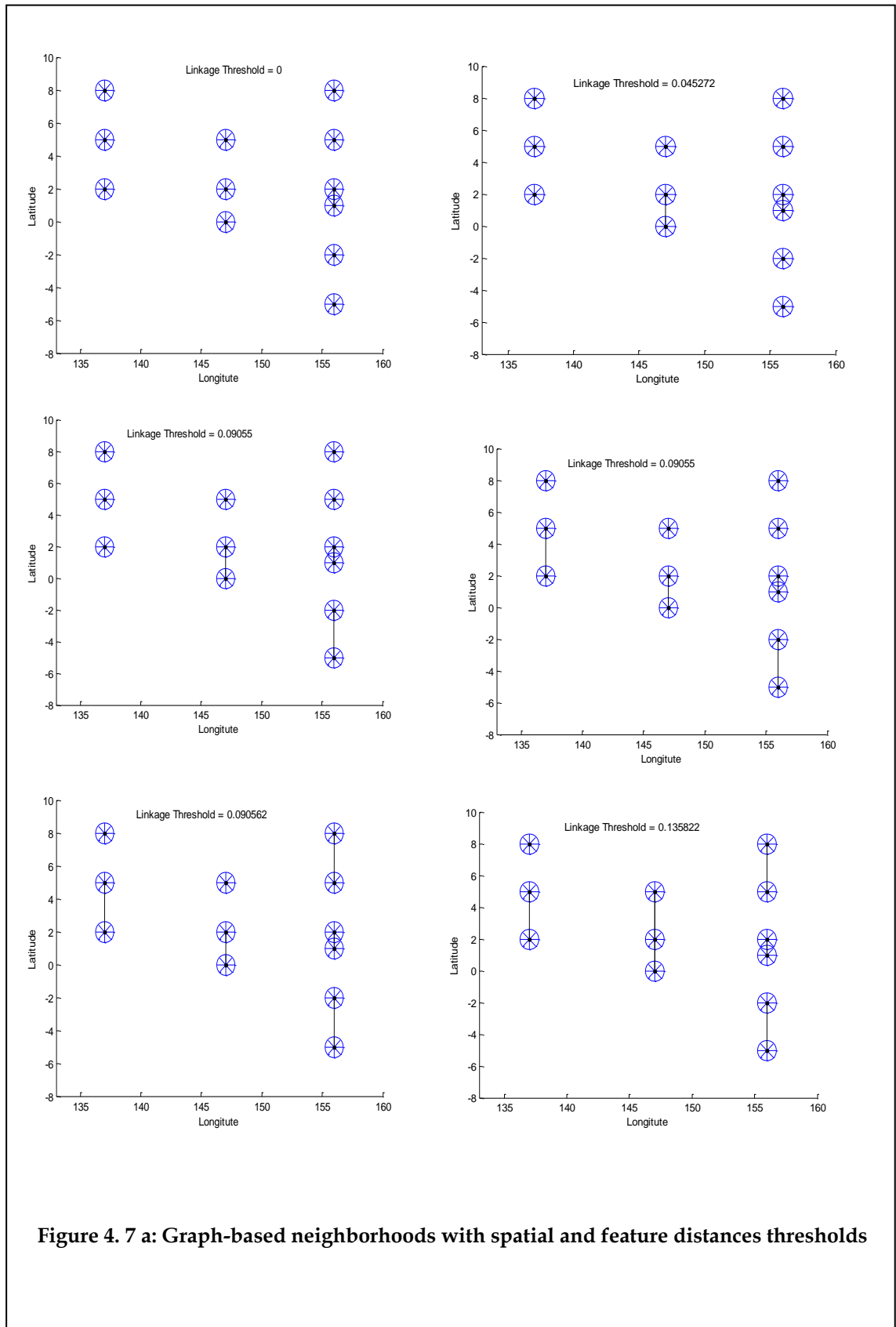
Figure 4. 7 a: Graph-based neighborhoods with spatial and feature distances thresholds

Figure 4. 7b: Graph-based neighborhoods with spatial distance thresholds (Continued).
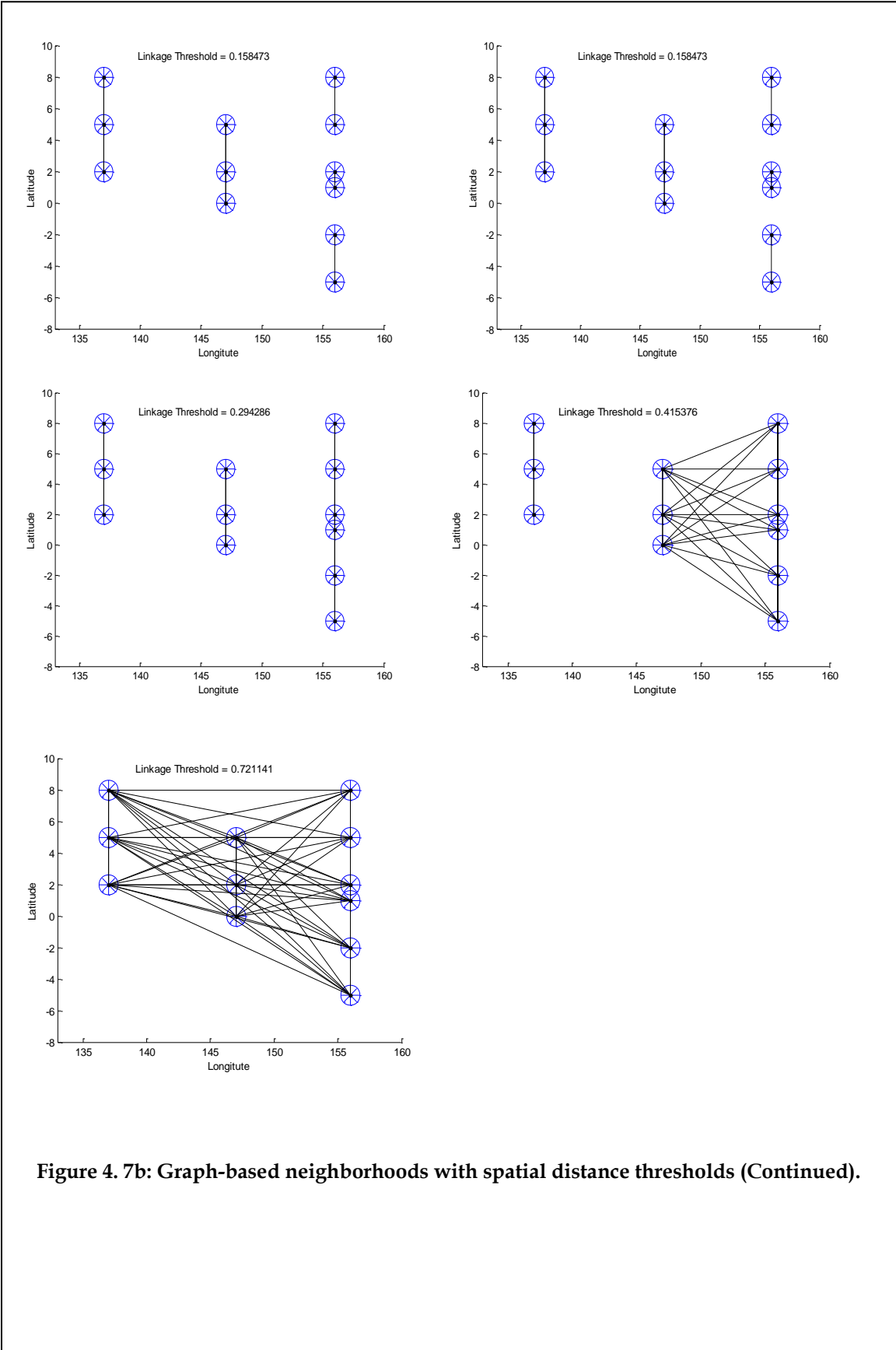
The iterations of the hierarchical algorithm are shown in Fig 4.7a and Fig 4.7b. Initially each sensor is considered as a single cluster, called singleton cluster and thus the total number of sensors and clusters both initially are 12. The clusters are agglomerated depending on the threshold distances, as shown in Table 4.8.

4.3.8.6 The Validation of Clustering Sensor Data

The cophenetic correlation coefficient is a measure used to determine the closeness of data and the cluster. The cophenetic correlation coefficient for a cluster tree is defined as the linear correlation coefficient between the cophenetic distances obtained from the tree (dendrogram), and the original distances (or dissimilarities) used to construct the tree. Thus, it is a measure of how faithfully the tree represents the dissimilarities among the observations. The closer the cophenetic correlation coefficient c gets to 1, the more accurately clustering solution reflects the data. The cophenetic correlation between Z and Y is defined as:

$$c = \frac{\sum_{i<j}(Y_{ij} - y)(Z_{ij} - z)}{\sqrt{\sum_{i<j}(Y_{ij} - y)^2 \sum_{i<j}(Z_{ij} - z)^2}}$$

where, Yij is the distance between object i and j in y

Zij is the cophenetic distance between objects i and j, from z

y and z are the averages of Y and Z, respectively

Y is the distance matrix

Z is the cophenetic distance matrix

In general, *C* = cophenetic (*Z*, *Y*) computes the cophenetic correlation coefficient for the hierarchical cluster tree represented by *Z*, where *Z* = linkage(*Y*). The linkage function creates a hierarchical cluster tree, using linkage algorithm. The input matrix, *Y*, is the distance matrix of length m(m-1)/2 where m is the number of sensors.

4.3.8.7 Dendrogram for Clustering Sensors with Similar SST

cophnet (c) = 0.8938
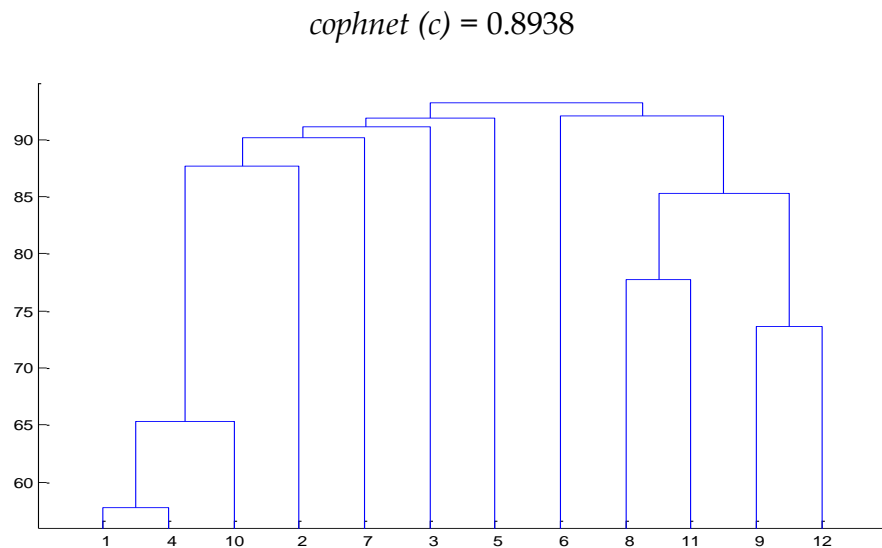
Figure 4. 8: A dendrogram for clustering sensors which have similar SST values

The dendrogram is created by grouping the sensors with the similar SST values by using the same procedure, single-linkage hierarchical clustering. The leaves of the dendrogram represent the sensors, and the length of the paths between

leaves represents the feature distance between the sensors. The value of cophenetic coefficient c = 0.8938 implies the clusters closely reflects the dataset.

4.3.8.8 The Dendrogram for Clustering Using the Weighted Algorithm

The dendrogram in Fig 4.9 is created by grouping the sensors that are spatially close and have similar SST values by using the extended single-linkage hierarchical clustering where weighted matrix = $s1 * 0.3 + s2 * 0.7$; $s1$ and $s2$ are the spatial and feature distances of the sensors, respectively.

O	T	P	D	
T1	H	L	Y	
T2	H	H	N	TIS
T3	L	L	N	
T4	L	H	N	

cophenet (c)= 0.7933
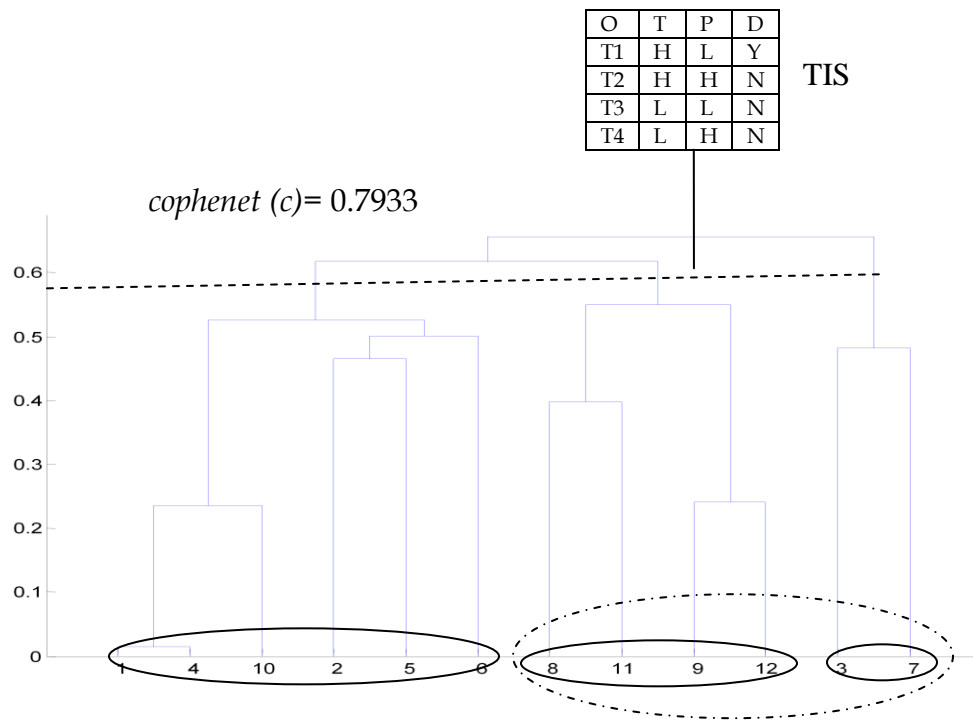
Figure 4.9: The dendrogram for clustering sensors which are spatially close and have similar SST values

We associate more weight on feature distance than spatial distance since the objective is to discover spatio-temporal patterns from the time series of the sensor dataset where feature values are similar. The leaves of the dendrogram represent the sensors and the length of the paths between the leaves denotes the weighted sum of spatial and feature distances between sensors. The value of cophenetic coefficient c = 0.7933 implies high-quality clustering. A cut in the dendrogram is made at a threshold > 0.6 and two clusters are obtained, shown by two big circles. The sensors within the circle (cluster), denoted by the dotted line, are only considered for the current research.

4.3.9 Data Discretization

Once we determine the sensors that are spatially close and have similar SST values, we can create temporal information system from the time series of SST obtained from the selected sensors. However, real time SST data is a continuous data and cannot be used in rough set based rule induction or decision tree generation since it is unlikely that these values match with the values of unseen, test data. Rough set theory has been mainly developed for nominal feature spaces (categorical or symbolic values). In order to use numerical values in RS either we need to convert numerical data into symbolic data through discretization before rule induction or both at the same time (Grzymala-Busse, 2003).

Discretization algorithms can be broadly classified as supervised and unsupervised discretization. In the supervised method, we consider all attributes as well as the information about decision or concept membership while the unsupervised algorithm only operates on the attributes but not on decisions (Grzymala-Busse, 2007). An unsupervised method is a blind method where no prior information is available; it solely depends on the distribution of attribute values. On the other hand, a supervised method uses domain knowledge and a priori information (I. U. Sikder, 2003). If the algorithm works on all attributes at the same time, then it is called global; otherwise it is local one i.e. one attribute at a time. Some well-accepted discretization algorithms are those of (Fayyad & Irani, 1993) and (Dougherty, Kohavi, & Sahami, 1995):

Unsupervised discretization method:

- Equal frequency binning

- Equal width binning

- Clustering

Supervised discretization method:

- Entropy-based

- Purity-based

We used the cluster, marked by a dotted line in Figure 4.9, which consists of the sensors s_3, s_7, s_8, s_9, s_{11} and s_{12}. We reduced the number of continuous values of SST by dividing the range of the attribute SST into intervals and assigning a label

to each interval. The division was based on several global cuts, performed only once for all attributes (sensors), and the cuts are at the mean value (μ), at the first standard deviation (σ) on the both sides of the mean value, and at 1.5σ on the both sides of mean value. Then interval labels are used to encode the actual attribute values, shown in Table 4.9. Thus, the continuous time series of sea surface temperature recorded by these sensors are discretized into five intervals as follows:

T($-\infty$, μ - $3\sigma/2$[;] μ - $3\sigma/2$, μ - σ[;] μ - σ, μ [;] μ, μ + σ[;] μ + σ, μ + $3\sigma/2$[;] μ + $3\sigma/2$, ∞)

The intervals are numbered from 1 to 5, respectively where mean μ = 29.26415, standard deviation σ = 0.655136.

Table 4. 9: Discretization intervals and the labels

Interval	Range	Label
($-\infty$, μ - $3\sigma/2$)	0 - 28.28145	1
(μ - $3\sigma/2$, μ - σ)	28.28145 - 28.60901	2
(μ - σ, μ)	28.60901 - 29.26415	3
(μ, μ + σ)	29.26415 - 29.91929	4
(μ + σ), μ + $3\sigma/2$)	29.91929 - 30.24685	5

Table 4.10 and Table 4.11 show the continuous values of SST within a cluster (before data discretization) and the discretized values of SST within a cluster (after discretization), respectively.

Table 4. 10 : Sea surface temperature readings within a cluster before discretization

sst2n137e_hr	sst5n137e_hr	sst5n147e_hr	sst5n156e_hr	sst8n137e_hr	sst8n156e_hr
28.88	28.24	29.12	29.16	27.65	28.27
28.87	28.25	29.15	29.17	27.68	28.29
29.01	28.25	29.18	29.19	27.72	28.31
29.12	28.32	29.22	29.2	27.78	28.33
29.15	28.47	29.24	29.19	27.85	28.33
29.34	28.53	29.29	29.19	27.92	28.34
29.26	28.55	29.28	29.19	28	28.34
29.4	28.55	29.27	29.19	28.03	28.32
29.33	28.58	29.25	29.19	28.01	28.3
29.18	28.6	29.22	29.19	27.91	28.29
29.15	28.55	29.2	29.2	27.85	28.28
…	…	…	…	…	…

Table 4. 11: Sea surface temperature readings within a cluster after discretization

sst2n137e_hr	sst5n137e_hr	sst5n147e_hr	sst5n156e_hr	sst8n137e_hr	sst8n156e_hr
3	1	3	3	1	1
3	1	3	3	1	2
3	1	3	3	1	2
3	2	3	3	1	2
3	2	3	3	1	2
4	2	4	3	1	2
3	2	4	3	1	2
4	2	4	3	1	2
4	2	3	3	1	2
3	2	3	3	1	2
3	2	3	3	1	1
…	…	…	…	…	…

4.3.10 Temporal Information System of Rough Sets

Once discretized, the nominal data set can be directly used in rough set based

data mining. The output of each global cut $T(-\infty, \mu - 3\sigma/2[;] \mu - 3\sigma/2, \mu - \sigma[;] \mu - \sigma,$

μ [;] μ, μ + σ[;] μ + σ, μ + 3σ/2[;] μ + 3σ/2, ∞) is used as the input of the temporal

information system in rough set shown in Table 4.12a.

The numeric dataset is created by using the following function:

$$f(r) = \begin{cases} n, & \text{if } r = \lceil s(t) \rceil = \min\{n \in Z \mid n \geq s(t)\} \end{cases}$$

where, Z is the set of all integers, $s(t)$ is the continuous value of the sensor

reading, and n is the encoded value of the sensor reading. Each sensor value $s(t)$

in Table 4.10 is encoded by the above function, and a temporal information

system for numeric data is created as shown in Table 4.12b.

The data, capable of handling outlier data, is created by the following function:

$$f(r) = \begin{cases} 1, & s(t) \geq r \\ 0, & s(t) < r \end{cases}$$

where $s(t)$ is the continuous value of the sensor reading and r is the mean value

calculated from the continuous sensor readings in Table 4.10.

Whenever the sensor reading $s(t)$ is greater than or equal to the threshold (r), the

continuous value $s(t)$ is encoded as 1, otherwise $s(t)$ is encodes as 0. Once we

determine

Table 4. 12a, b, c: Temporal information system within a cluster for discrete, numeric, and Boolean dataset

6 4368	A[0]	A[1]	A[2]	A[3]	A[4]	A[5]
1	3	1	3	3	1	1
2	3	1	3	3	1	2
3	3	1	3	3	1	2
4	3	2	3	3	1	2

5	3	2	3	3	1	2
6	4	2	4	3	1	2
7	3	2	4	3	1	2
8	4	2	4	3	1	2
9	4	2	3	3	1	2
10	3	2	3	3	1	2
11	3	2	3	3	1	1
…	…	…	…	…	…	…

6 / 4368	A[0]	A[1]	A[2]	A[3]	A[4]	A[5]
1	29	29	30	30	28	29
2	29	29	30	30	28	29
3	30	29	30	30	28	29
4	30	29	30	30	28	29
5	30	29	30	30	28	29
6	30	29	30	30	28	29
7	30	29	30	30	28	29
8	30	29	30	30	29	29
9	30	29	30	30	29	29
10	30	29	30	30	28	29
11	30	29	30	30	28	29
…	…	…	…	…	…	…

6 / 4368	A[0]	A[1]	A[2]	A[3]	A[4]	A[5]
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0
6	1	0	1	0	0	0
7	0	0	1	0	0	0
8	1	0	1	0	0	0
9	1	0	0	0	0	0
10	0	0	0	0	0	0
11	0	0	0	0	0	0
…	…	…	…	…	…	…

the outlier and encode the sensor readings in Table 4.10 by employing the above function, a temporal information system can be created (shown in Table 4.12c) which can then be directly used for outlier data mining.

4.4 Spatio-temporal Pattern and Rule Generation

The algorithm for template and rule generation explained in Chapter 3 is tested on the discrete, numeric, and outlier datasets. The experiment is repeated also for several different template qualities and window sizes. However, for the sake of space, all results are not shown in this dissertation.

4.4.1 Spatio-temporal Patterns Generation

4 214	Interval	Code	Template
1	0	0	
2	1	1	a1=29 & a2=30 & a3=30 & a4=28 & a5=29
3	2	2	a1=29 & a2=30 & a3=30 & a5=29
4	9	3	a2=30 & a3=30 & a5=29
5	10	4	a2=30 & a3=30 & a4=28 & a5=29
6	14	5	a0=30 & a2=30 & a3=30 & a4=28 & a5=29
7	16	4	a2=30 & a3=30 & a4=28 & a5=29
8	17	3	a2=30 & a3=30 & a5=29
9	18	6	a0=30 & a1=30 & a5=29
10	19	7	a0=30 & a1=30 & a2=29 & a4=28 & a5=29
11	20	8	a0=30 & a2=29 & a5=29
12	22	9	a0=30 & a3=30 & a5=29
13	23	0	
14	25	10	a1=30 & a5=29
15	27	11	a0=30 & a1=30 & a2=30 & a5=29
16	28	12	a1=30 & a2=30 & a5=29
17	29	10	a1=30 & a5=29
…	…	…	…
210	32 8	0	
211	331	85	a0=30 & a2=30
212	332	107	a2=30 & a4=30
213	334	68	a1=30 & a2=30 & a4=30
214	335	69	a0=30 & a1=30 & a2=30 & a4=30

Table 4. 13 Spatio-temporal Templates for numeric data: Window size = 26, Step size = 13, Quality = 90

Table 4.13 shows a series of homogeneous spatio-temporal patterns generated by scanning the temporal information system for the numeric dataset shown in Table 4.12a within a window of 26 hours in each interval. The threshold used for obtaining these patterns is a quality of 90%, which means that the pattern must appear at least 90 % of the time the size of the window. Then the upper and the lower bounds of the window where the patterns satisfy the quality requirement are determined. It is also possible that there is no interesting pattern in an interval. The windows are refreshed after 13 hours and the same procedure of discovering interesting patterns continues until the last element of the TIS is scanned. By scanning the TIS of 4368 objects, we obtained only 107 unique spatio-temporal patterns, determined from the unique number called *code* associated with each pattern.

Table 4.13 provides much interesting information, such as the number of unique templates in the TIS, probability mass function of the templates, mean template density, etc. The probability mass function for the templates generated from the numeric data for window size = 26 hours, step = 13 hours, and the quality = 90% is shown in Fig. 4.10. The significance of the probability mass function will be evident when the uncertainty management issues come in the following sections.
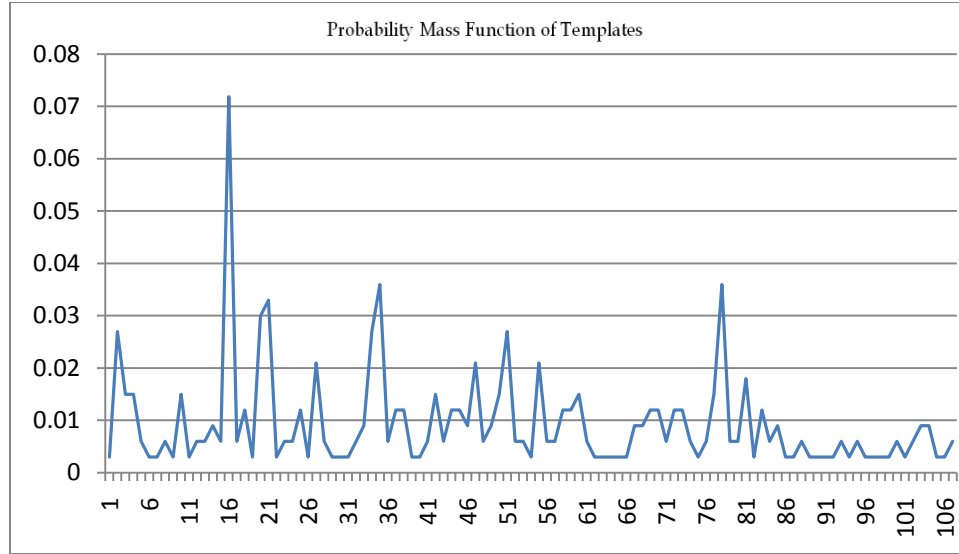
Figure 4. 10: Probability mass function of templates

4.4.2 Multi-valued Decision System

Table 4. 14 An Multi-valued Decision System for numeric data: Window size = 26, Step size = 13, Quality = 90

3 187	A[0]	A[1]	Decision
1	{29}	{30}	{2}
2	{5}	{4}	{3}
3	{44}	{37, 43}	{3}
4	{32}	{33}	{3}
5	{1}	{2}	{3}
6	{2}	{3}	{4}
7	{4}	{5}	{4}
8	{3}	{4}	{5}
9	{4}	{3}	{6}
10	{3}	{6}	{7}
11	{6}	{7}	{8}
12	{7}	{8}	{9}
13	{59, 65}	{66}	{10}
14	{8}	{9}	{10}
15	{11}	{12}	{10}
16	{9}	{10]	{11}
17	{10}	{13}	{12}
18	{10}	{11}	{12}
19	{42}	{43}	{13}

20	{12}	{10}	{13}
21	{13}	{12}	{14}
22	{20}	{21}	{14}
…	…	…	…

The multi-valued decision table, shown in Table 4.14, is created from the series of the templates generated in the previous section. The table has 187 cases and 3 attributes. While constructing the decision table, only the templates at time (t-1) and (t–2) are considered as the condition attributes for the decision making at time t. This table shows that the multi-valued decision systems can have multiple values for an attribute i.e. it supports parallel templates. The multi-valued decision system is used to generate temporal rules for decision making.

4.4.3 Rough Set Rule Induction

Table 4. 15 Rough set Rule induction for numeric data set: Window size = 26, Step size = 13, Quality = 90

3 264	Rule	Support	Match
1	(30ca1) => D=2	1	1
2	(29ca0) => D=2	1	1
3	(5ca0) => D = 3	1	1
4	(44ca0)&(37ca1) => D=3	1	1
5	(37ca1)&(43ca1) => D=3	1	1
6	(44ca0)&(43ca1) => D=3	1	1
7	(33ca1) => D=3	1	1
8	(32ca0) => D=3	1	1
9	(1ca0) => D=3	1	1
10	(2ca0)&(3ca1) => D=4	1	1
11	(5ca1) => D=4	1	1
12	(3ca0)&(4ca1) => D=5	1	1
13	(4ca0)&(3ca1) => D=6	1	1
…	…	…	…
171	(51ca0)&(69ca0) => D=68	1	1
172	(51ca0)&(51ca1) => D=68	1	1

17 3	(68ca1) => D=69	3	3
…	…	…	…
260	(68ca0) => D=69	1	2
261	(68ca0) => D=70	1	2
262	(58ca0)&(81ca1) => D=78	1	2
263	(58ca0)&(81ca1) => D=81	1	2
264	(81ca0)&(73ca1) => D=89	1	2

Before generating the rules from the multi-valued decision system, an important step is to extract and eliminate the redundant patterns which are not useful for decision making. The next important step is to find the reduct, the minimal set of attributes to preserve the equivalence relation. A rough set based heuristic algorithm is used to determine the reducts by employing the degree of dependency as a measure. Once the reducts are computed, the temporal rules are induced by associating the values from the reduced table. Table 4.15 lists the temporal rules generated from the multi-valued decision system created in the previous section. The table shows that 264 temporal rules are generated from the temporal information system of 4368 observations, which implies that a significant data reduction can be achieved by employing RS rule mining on sensor data.

4.4.4 Rough Set Rule Validation and the Confusion Matrix

Table 4. 16 Rough set Rule Validation: Confusion Matrix (Numeric data set)

			Predicted Values										
		56 53	…	{42,50}	{42}	{43}	{44}	{45}	{46}	{47}	{48}	{49}	…	Accuracy
Actual Values	…	…	…	…	…	…	…	…	…	…	…	…	…
	{42,50}	…	0	0	0	0	0	0	0	0	0	…	0
	{42}	…	0	2	0	0	0	0	0	0	0	…	1
	{43}	…	0	0	1	0	0	0	0	0	0	…	1
	{44}	…	0	0	0	1	0	0	0	0	0	…	1
	{45}	…	0	0	0	0	1	0	0	0	0	…	1
	{46}	…	0	0	0	0	0	1	0	0	0	…	1
	{47}	…	0	0	0	0	0	0	1	0	0	…	1
	{48}	…	0	0	0	0	0	0	0	1	0	…	1
	{49}	…	0	0	0	0	0	0	0	0	1	…	1
	Accuracy	…	Unde-fined	1	0.5	1	1	1	1	1	1	…	0.897

Instead of sending the entire temporal information system, only the rules generated at the cluster head are periodically sent to the sink. Thus it is very important to ensure that these rules represent the temporal information system before they are sent to the sink. In other words, we should perform the fidelity test on the rules in order to verify how faithfully the rules represent the aggregated data at the cluster head. The fidelity test on the rules as a measure of rule quality is suggested by many researchers (Bologna, 2002a, 2002b; Craven & Shavlik, 1999).

The datasets are randomly partitioned into two disjoint datasets, and tests are performed on the both sets to ensure the rule fidelity and consistency. By

constructing the confusion matrix where each column denotes the predicted templates, each row represents the actual templates, the diagonal elements correspond to the correctly classified templates, and the off-diagonal elements signify misclassified templates, we perform the rule validation and determine the accuracy of classification as described in the following section. The confusion matrix shown in Table 4.16 is created from the numeric dataset.

4. 4. 5 The Accuracy Estimation of Rules

The accuracy of the classifier is defined as the ratio of the correctly classified templates in a class to the total number of templates in the class that are classified. Thus, the accuracy of classification (α) is computed as:

$\alpha = \sum X_{ii}/N$

where,

$\sum X_{ii}$ is the number of diagonal elements that are classified correctly.

N is the number of all elements that are classified.

The accuracy of the classifier shown in the confusion matrix is 0.897 where we use numeric data set, template quality = 90%, window size = 26 hrs, and step size = 13 hrs. The accuracy estimation is performed on different datasets and the results are summarized in the following section.

4.4.6 Comparison of Rule Mining in Discrete, Numeric, and Boolean Datasets

Table 4.17: A comparison of rule mining for discrete, numeric, and outlier datasets

	Discrete data	Numeric data	Boolean data
Temporal Information System: • No. of Attributes • No. of Objects	6 4368	6 4368	6 4368
Temporal Templates: • No. of Templates • Parameters of templates: • Quality • Window Size • Step	151 90 26 13	108 90 26 13	93 90 26 13
Multi-valued Decision System: • No. of Attributes • No. of Objects	3 191	3 187	3 159
Temporal Rules: • No. of Temporal Rules • Consistent Rules • Inconsistent Rules	317 315 2	264 250 14	222 195 27
Accuracy: • 1st data set: • 2^ndataset:	0.969 0.848	0.897 0.891	0.942 0.778

Table 4.17 summarizes the results regarding the pattern generation and rule extraction from discrete, numeric, and outlier datasets. For the same template quality = 90%, window size = 26 hrs, and step size = 13 hrs, we obtain 151 unique templates from discrete data set, 108 unique templates from numeric dataset, and 93 unique templates from Boolean dataset. The potential cause for this variation may be attributed to the number of discrete levels for each data set. For instance we used 5 levels to discretize discrete dataset, 4 levels for the numeric dataset, and 2 levels for the Boolean or outlier data set. For numeric data, we do not explicitly use 4 levels but we use ceiling function to encode the sensor readings. Since the SST values change within 28-31 degree, the encoding based on ceiling function produces only 4 levels. Because of the combinatorial effect of different encoded values, we get more unique templates from the discrete dataset, compared to the other data sets. Consequently, the number of objects in the multi-valued decision system and the number of temporal rules are also higher for discrete dataset than numeric and Boolean datasets. However, there is no significant difference in average estimated accuracy for these datasets.

4.4.7 Data Compression in Spatio-temporal Pattern Generation

The pattern-based data aggregation technique proposed in the dissertation eliminates data redundancy from the sensor reading and extracts the rules at the cluster head nodes. It prevents the redundant data transmission from the cluster head nodes to the sink by intelligently determining the features that are not

important for decision making and eliminating those features locally at cluster heads. Thus it has the potential to provide significant reduction in data communication as well as energy, given that the data transmission is the major cause of energy consumption. In the following section, the data compression is estimated for discrete, numeric, and Boolean datasets by varying the quality of the templates.

4.4.7.1 Data Compression vs. Template Quality

Fig. 4.11 shows that a significant data compression can be achieved by employing the proposed data aggregation method in the context of constantly evolving continuous data, such as WSN data. This result is based on the test performed on all three datasets, namely symbolic, numeric, and Boolean, and the template quality values ranging from 20% to 99%.
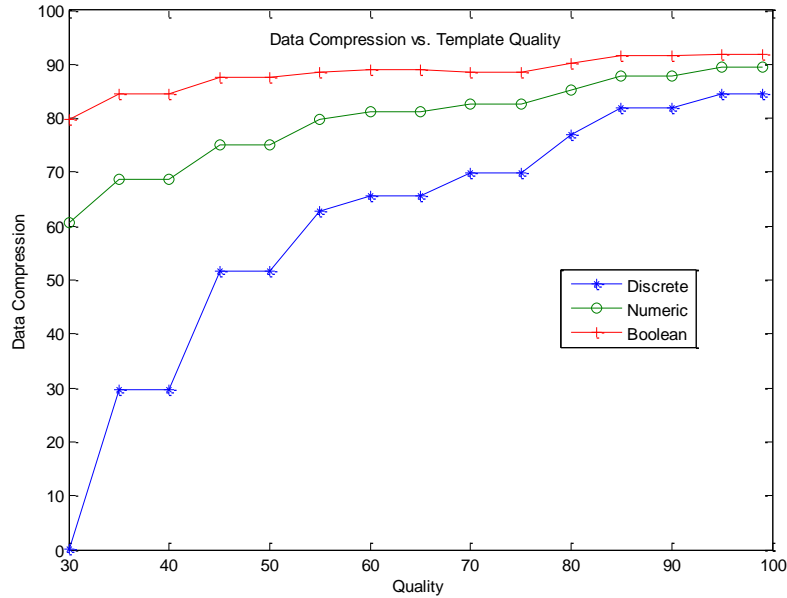
Figure 4. 11: Data compression vs. template quality

 Fig. 4.11 shows that higher data compression is achieved with the increase of template quality irrespective of the datasets. This result is expected since fewer unique templates that can satisfy the increased quality threshold are sent to the sink. Also, the result shows that the highest data compression is possible with the Boolean datasets compared to the other datasets. This can be explained by the combinatorial effect of a fewer discrete levels which is 2 in the Boolean data set, compared to the 4 discrete levels for numeric and 5 discrete levels for discrete datasets. The compression is estimated by the number of representative rules generated from the spatio-temporal patterns divided by the number of objects in the temporal information system since only the rules are sent to the sink.

4.4.7.2 Compression Loss Due to Vacuous Templates
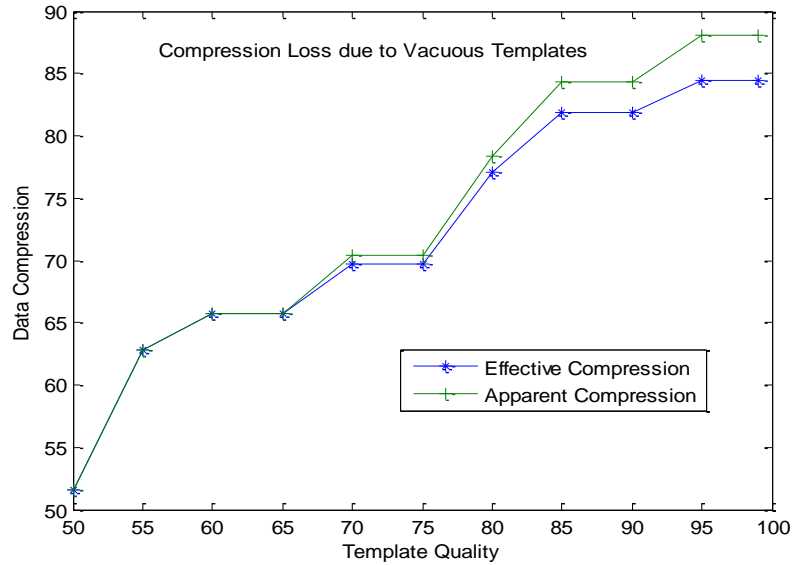
Figure 4.12: Data compression loss due to vacuous templates

Fig. 4.12 shows the effect of the data compression loss due to vacuous templates for discrete data sets. In order to obtain the actual data compression resulting from the data aggregation at the cluster head it is important to compute the compression loss due to vacuous templates that do not provide any information in the context of data aggregation. Once we know the compression loss due to vacuous templates, the effective data compression can be computed by subtracting the compression loss from the apparent data compression. Fig 4.12 shows the apparent and the effective data compression at different quality levels for discrete datasets.

4.5 Uncertainty Management in WSN

Uncertainty management is a growing research area (Klir & Wireman, 1999). The interest in uncertainty management will grow as we continue to develop complex systems and the technologies to analyze the complex systems for critical decision making. WSN is an example of a complex system that has a potential to observe and understand large-scale, real-world phenomena at a fine spatio-temporal resolution and help us with critical decision making. However, severe resource constraints, frequent changes in the dynamics of the environment, and a large number of unattended sensors create uncertainties in WSN. Thus it is important to characterize and quantify the uncertainties that limit the potential applications of WSN in real life problems. In the following sections, several uncertainty measures are defined and quantified in the context of the template generation and data aggregation in WSN. The sensitivity analysis is also performed on the uncertainty measures to determine the tradeoff among them.

4.5.1 The Formal Definitions of the Uncertainty Measures in WSN

There are several uncertainty measures that engineers or scientists find useful to quantify different categories of uncertainties, such as entropy, nonspecificity, vagueness, etc. The following uncertainty measures are defined in the context of template generation and data aggregation in WSN:

- Entropy-based uncertainty - results from conflict among evidential claims.

- Nonspecificity - stems from imprecision associated with the sizes or cardinalities of relevant sets of alternatives.

- Fuzziness or vagueness results from the imprecision in boundaries of sets.

- Unique templates – determine the number of unique spatio-temporal patterns in a given time period.

- Template Recurrence – determines the frequency of the occurrences of a particular spatio-temporal pattern in a dataset for a given time period.

- Template Recurrence Variability - measures the variation of the frequency of occurrences of a particular spatio-temporal pattern in a dataset.

- Template Vacuity – counts vacuous template that implies there is no interesting patterns or templates in dataset.

Definition 4.1 The entropy-based uncertainty $H(x)$ in the temporal information system $S = \{x_1, x_2, \ldots, x_n, s_1, s_2, \ldots, s_n\}$ stemming from the templates $\Im = \{\Im_1, \Im_2, \ldots, \Im_k\}$ is defined as:

$$H(x) = \sum_{i=1}^{k} p_i \log_2 \left(\frac{1}{p_i}\right)$$

where x_1, x_2, ..., x_n are the sensor readings at time t_1, t_2, ..., t_n; s_1, s_2, ..., s_n are the set of sensors within the cluster of interest; and p_1, p_2, ... p_k denote the probabilities of the template $\Im_1$, $\Im_2$, ... , $\Im_k$, respectively, so that $\sum_{i=1}^{k} p_i = 1$.

Definition 4.2 The nonspecificity-based uncertainty *N(m)* in the temporal information system S = {x_1, x_2, ..., x_n, s_1, s_2, ..., s_n} stemming from the templates $\Im = \{\Im_1, \Im_2,...., \Im_k\}$ is defined as:

$$N(m) = \sum_{i=0}^{2^k} m(A_i) \log_2(|A_i|)$$

where x_1, x_2, ..., x_n are the sensor readings at time t_1, t_2, ..., t_n; s_1, s_2, ..., s_n are the set of sensors within the cluster of interest; m_1, m_2, ... m_{2^k} denote the basic probability assignments of the subsets A_1, A_2, ..., A_{2^k} , respectively; A_1, A_2, ..., A_{2^k} are the subsets of the power set *P(A)* of the set of templates $\Im_1$ = $\Im_1$, $\Im_2$, ... , $\Im_k$; m is the basic probability assignment of the power set *P(A)* so that $\sum_{i=1}^{2^k} m(A_i) = 1$; $P(A) = \cup_{i=1}^{2^k} A_i$; and $(\forall B)(\exists P)(\forall A)$ $(B \in P \Leftrightarrow B \subseteq A)$ for A = {A_1, A_2, ..., A_{2^k}).

Definition 4.3 The vagueness-caused uncertainty (U_i) in the temporal information system S = {x_1, x_2, ..., x_n, s_1, s_2, ..., s_n} stems from the inconsistent rules, derived from the boundary region of approximation space, that cannot

certainly predict the causal relationships among the template sets $\Im = \{\Im_1, \Im_2, ..., \Im_k\}$ and can be approximated as:

$$U_i = \frac{I}{R}$$

where, I is the number of inconsistent rules derived from the boundary region of the concept and R is the total number of rules derived from the approximation space of positive, negative, and boundary regions.

Definition 4.4 The template-based uncertainty (U_T) in the temporal information system $S = \{x_1, x_2, ..., x_n, s_1, s_2, ..., s_n\}$ stemming from the templates $\Im = \{\Im_1, \Im_2, ..., \Im_k\}$ is defined as follows:

$$U_T = k$$

where, k is the number of unique templates ID generated from the information system S, defined by the set of sensor readings $x_1, x_2, ..., x_n$ at time $t_1, t_2, ..., t_n$ and set of sensors $s_1, s_2, ..., s_n$ and the templates must satisfy the condition ***Quality*** $\geq q_c.$; q_c is a predefined threshold for quality and the higher value of q_c may decrease the template-based uncertainty.

Definition 4.5 The mean template recurrence-based uncertainty ($\bar{F}$) in the temporal information system $S = \{x_1, x_2, ..., x_n, s_1, s_2, ..., s_n\}$ stemming from the

templates $\Im = \{\Im_1, \Im_2, ..., \Im_k\}$ and characterizes the mean frequency of the templates and is computed as:

$$\bar{F} = \frac{\sum_{i=0}^{k} F_i}{K}$$

where $x_1, x_2, ..., x_n$ are the sensor readings at time $t_1, t_2, ..., t_n$; $s_1, s_2, ..., s_n$ are the set of sensors within the cluster of interest; $F_1, F_2, ... F_k$ denote the frequencies of the templates $\Im_1$, $\Im_2$, ..., $\Im_k$, respectively; The higher values of $\bar{F}$ implies lower mean template based uncertainties of the S.

Definition 4.6 The mean template recurrence variability-based uncertainty ($\bar{R}$) in the temporal information system $S = \{x_1, x_2, ..., x_n, s_1, s_2, ..., s_n\}$ stemming from the templates $\Im = \{\Im_1, \Im_2, ..., \Im_k\}$ characterizes the mean variation of the templates from the average template frequency and is computed as:

$$\bar{R} = \sqrt{\frac{1}{K}\sum_{i=0}^{K}(F_i - \bar{F})^2}$$

where $x_1, x_2, ..., x_n$ are the sensor readings at time $t_1, t_2, ..., t_n$; $s_1, s_2, ..., s_n$ are the set of sensors within the cluster of interest; $F_1, F_2, ... F_k$ denote the frequencies of the templates $\Im_1$, $\Im_2$, ..., $\Im_k$, respectively; and $\bar{F}$ represents the mean frequency of the templates. The higher values of $\bar{R}$ implies higher mean template recurrence-based uncertainties of the S.

Definition 4.7 The vacuous template-based uncertainty (U_V) in the temporal information system $S = \{x_1, x_2, \ldots, x_n, s_1, s_2, \ldots, s_n\}$ stemming from the templates $\mathfrak{I} = \{\mathfrak{I}_1, \mathfrak{I}_2, \ldots, \mathfrak{I}_k\}$ characterizes the templates that have no interesting patterns and is computed as:

$$U_V = n'$$

where, $n' \leq N$ is the number of templates for which template ID = 0 and **Quality** $< q$, generated from the information system S, defined by the set of sensor readings $x_1, x_2, \ldots, x_n$ at time $t_1, t_2, \ldots, t_n$ and set of sensors $s_1, s_2, \ldots, s_n$ and; q is a predefined threshold for quality and in general, the higher value of q may increase the vacuous template-based uncertainty.

4.5.2 The Sensitivity Analysis of the Uncertainty Measures in WSN

After quantifying all the uncertainty measures that we define in the previous section, a sensitivity analysis is performed for each pair of uncertainty measures and the results are demonstrated in several graphs in the following subsections. This analysis is important for uncovering the tradeoff among the uncertainty measures.

4.5.2.1 Template Entropy vs. Window Size
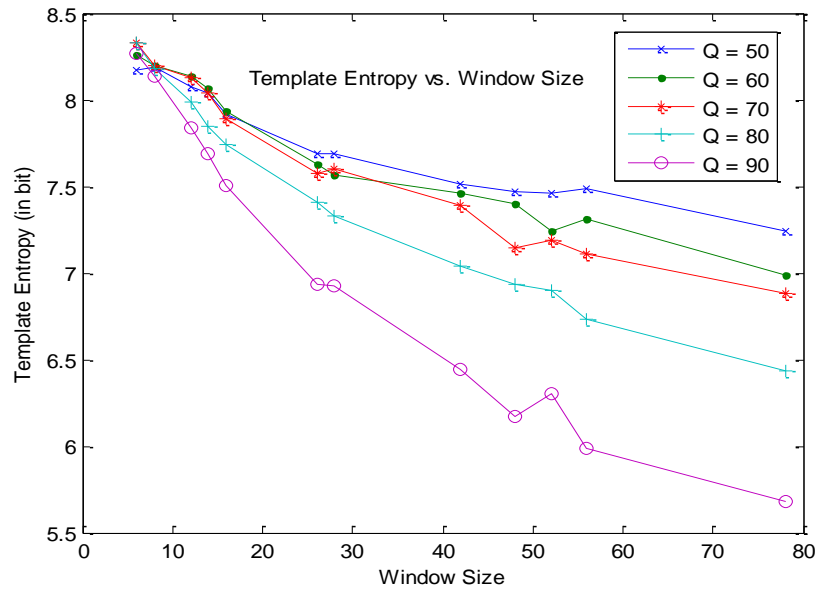
Figure 4.13: Template entropy vs. window size

Fig 4.13 demonstrates the change of entropy by varying the window size from 5 hours to 78 hours and the template quality from 60% to 90% in step of 10%. It clearly shows that the entropy decreases with the increase of window size irrespective of the template quality. However the decrease in entropy is drastic when the template quality is high. This result is expected based on the formal definition of entropy that we provide in the prior section and Chapter 3. The entropy depends on the number of templates and the split of each template. Thus the decreased number of wider templates brings the entropy down. This effect is even more drastic for high quality fewer templates.

4.5.2.2 Template Nonspecificity vs. Window Size
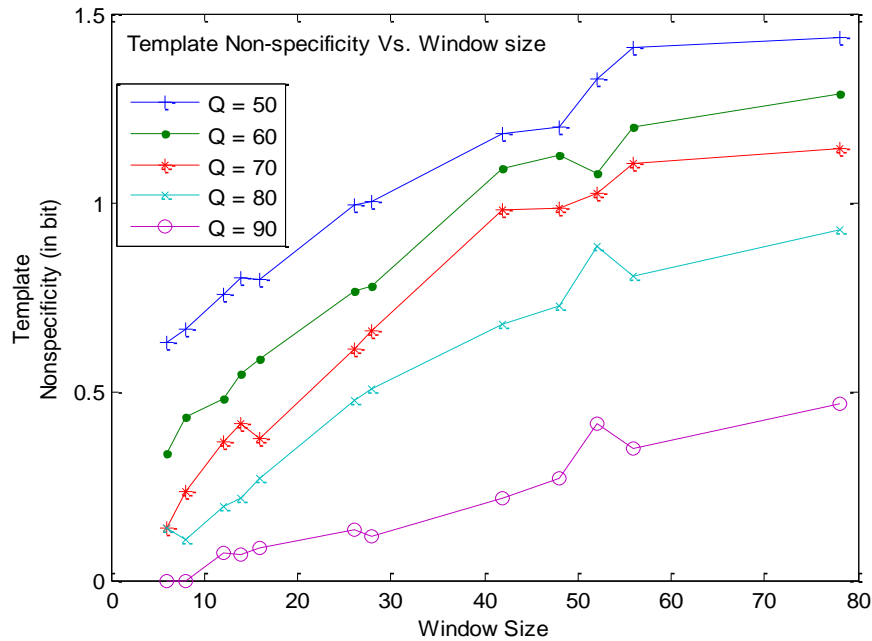
Figure 4.14: Template nonspecificity vs. window size

Fig 4.14 shows the change of nonspecificity by varying the window size from 5 hours to 78 hours and the template quality from 60% to 90% in step of 10%. It yields that the nonspecificity increases with the increase of window size irrespective of the template quality. However the increase in nonspecificity is drastic when the template quality is low. This result is expected based on the formal definition of nonspecificity that we provide in the prior section and Chapter 3. The nonspecificity depends on the number of parallel templates i.e. the templates that occur in the same interval and the basic probability assignment of each subset of the set of parallel templates. Thus the increased number of low quality templates increases the nonspecificity. This effect is even

more drastic when the templates are wider. Although the number of templates is decreasing with the increase of window size, the probability assignment for the wider template is higher than the narrower templates because of the less variability in the wider templates. Thus the nonspecificity increases with the increase of window size.

4.5.2.3 The Boundary Region based Uncertainty vs. Window Size
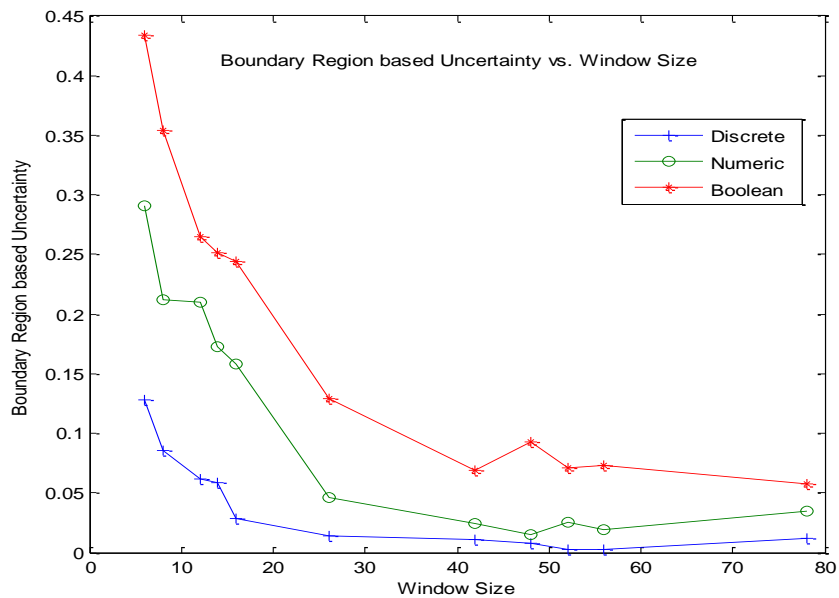

Figure 4.15: The Boundary region based uncertainty vs. window size

Fig 4.15 demonstrates the change of boundary-region-based uncertainty or vagueness-caused uncertainty by varying the window size from 5 hours to 78 hours for discrete, numeric, and Boolean datasets. It illustrates that the boundary region based uncertainty decreases with the increase of window size irrespective of the datasets. However the decrease in the uncertainty is drastic for Boolean datasets. This result is expected based on the formal definition of boundary-region based or vagueness-caused uncertainty that we provided in the prior section and Chapter 3. This type of uncertainty depends on the number of templates and the granularity in datasets. The Boolean dataset has less granularity and hence more uncertainty because of the fewer number of discrete levels, which is 2, in contrast to 4 levels in numeric and 5 levels in discrete datasets. The increase in window size has a negative impact on the boundary region based uncertainty because of the fewer number of templates and consequently a fewer number of inconsistent rules induced from the boundary region.

4.5.2.4 The Other Uncertainty Measures vs. Window Size

Fig 4.16a demonstrates the sensitivity of the number of unique templates on window size for several different template qualities from 60% to 90% in step of 10%.
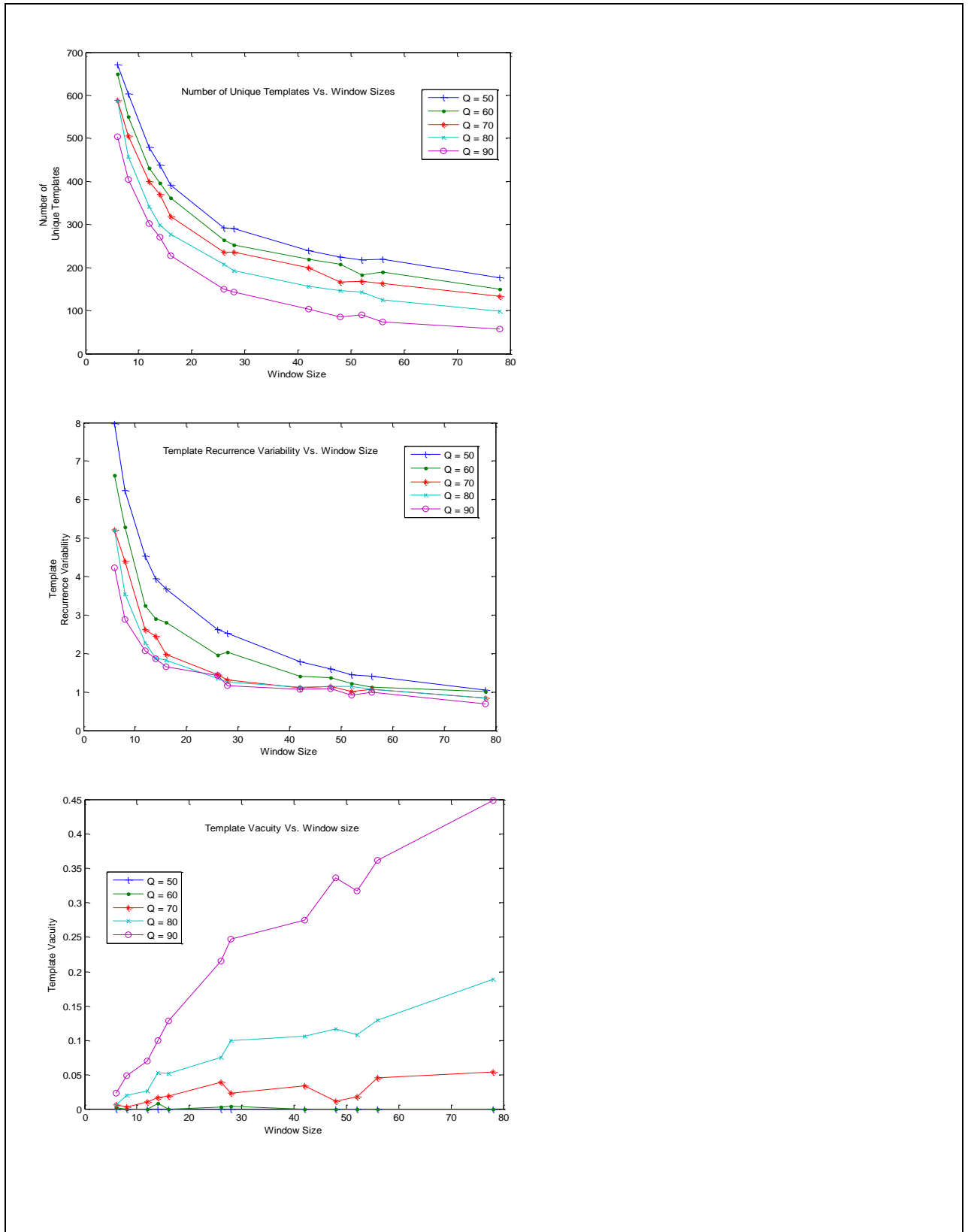
Figure 4.16: Other uncertainty measures vs. window size: a) number of unique templates b) template recurrence variability c) template vacuity

It clearly shows that the number of unique templates decreases with the increase of window size irrespective of the template quality. However a further decrease in template number is observed when the quality of the template increases. This result can be explained based on the formal definition of unique template that we provide in the prior section and Chapter 3. The increased window size requires the templates to be active on the window for a longer time and hence the template becomes wider and the number of templates reduces. For the same reason the number of unique templates decreases with the increase of template quality since a fewer number of templates satisfy the increased quality requirements.

Fig 4.16b demonstrates the change of the template recurrence variability by varying the window size from 5 hours to 78 hours and the template quality from 60% to 90% in step of 10%. It clearly yields that the templates recurrence variability decreases with the increase of window size irrespective of the template quality. However a further decrease in template recurrence variability is observed when the quality of the template increases. This result can be explained based on the formal definition of the template recurrence variability that we provide in the prior section and Chapter 3. The increased window size reduces the number of unique templates but increases the mean template recurrence and hence decreases template recurrence variability. For the same

reason, with the increase of template quality, the template recurrence variability decreases.

Fig 4.16c demonstrates the change of the number of the vacuous templates by varying the window size from 5 hours to 78 hours and the template quality from 60% to 90% in step of 10%. It clearly shows that the number of vacuous templates increases with the increase of window size irrespective of the template quality. However a drastic increase is observed when the quality of the template increases. This result can be explained based on the formal definition of the template recurrence variability that we provide in the prior section and Chapter 3. The increased window size imposes additional requirements and reduces the number of unique templates and hence increases the number of vacuous templates. For the same reason with the increase of template quality the number of vacuous templates increases.

4.5.2.5 H(T)/H(T)max vs. Template Quality
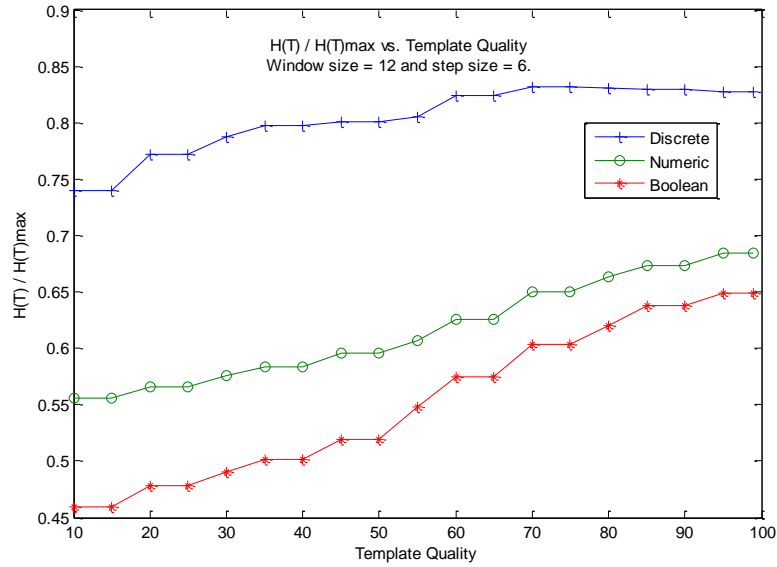
Figure 4.17: H(T)/H(T)max vs. template quality

Fig 4.17 demonstrates the sensitivity of the normalized entropy with the template quality for discrete, numeric, and Boolean datasets. It shows that the ratio H(T)/H(T)max tends to 1 with the increase of template quality irrespective of the type of datasets. However the normalized entropy is higher for discrete dataset in comparison to numeric and Boolean datasets. The general increase in normalized entropy can be explained as a consequence of the increased maximum entropy due to high quality fewer templates.

4.5.2.6 Template Nonspecificity vs. Template Quality
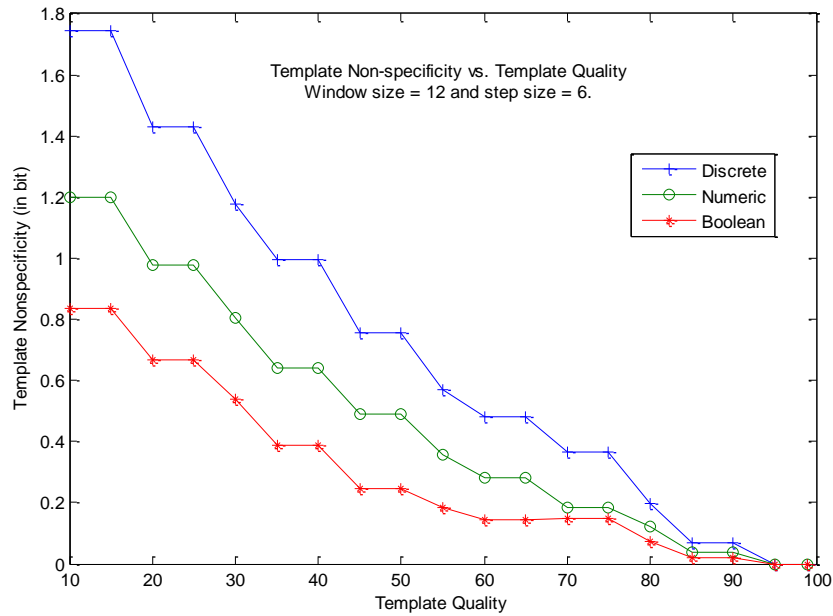
Figure 4.18: Template nonspecificity vs. template quality

Fig 4.18 demonstrates the sensitivity of the nonspecificity with the template quality for discrete, numeric, and Boolean datasets. It shows that the nonspecificity-based uncertainty decreases with the increase of template quality irrespective of the type of datasets. However the nonspecificity is higher for discrete dataset in comparison to numeric and Boolean datasets. The general trend of the decrease in nonspecifity can be explained as a consequence of high quality fewer templates. For the same reason, the nonspecificity is lower for the Boolean dataset which generates fewer numbers of templates due to a fewer number discrete levels, which is 2, compared to 4 levels in numeric, and 5 in discrete datasets.

4.5.2.7 The Other Uncertainty Measures vs. Template Quality
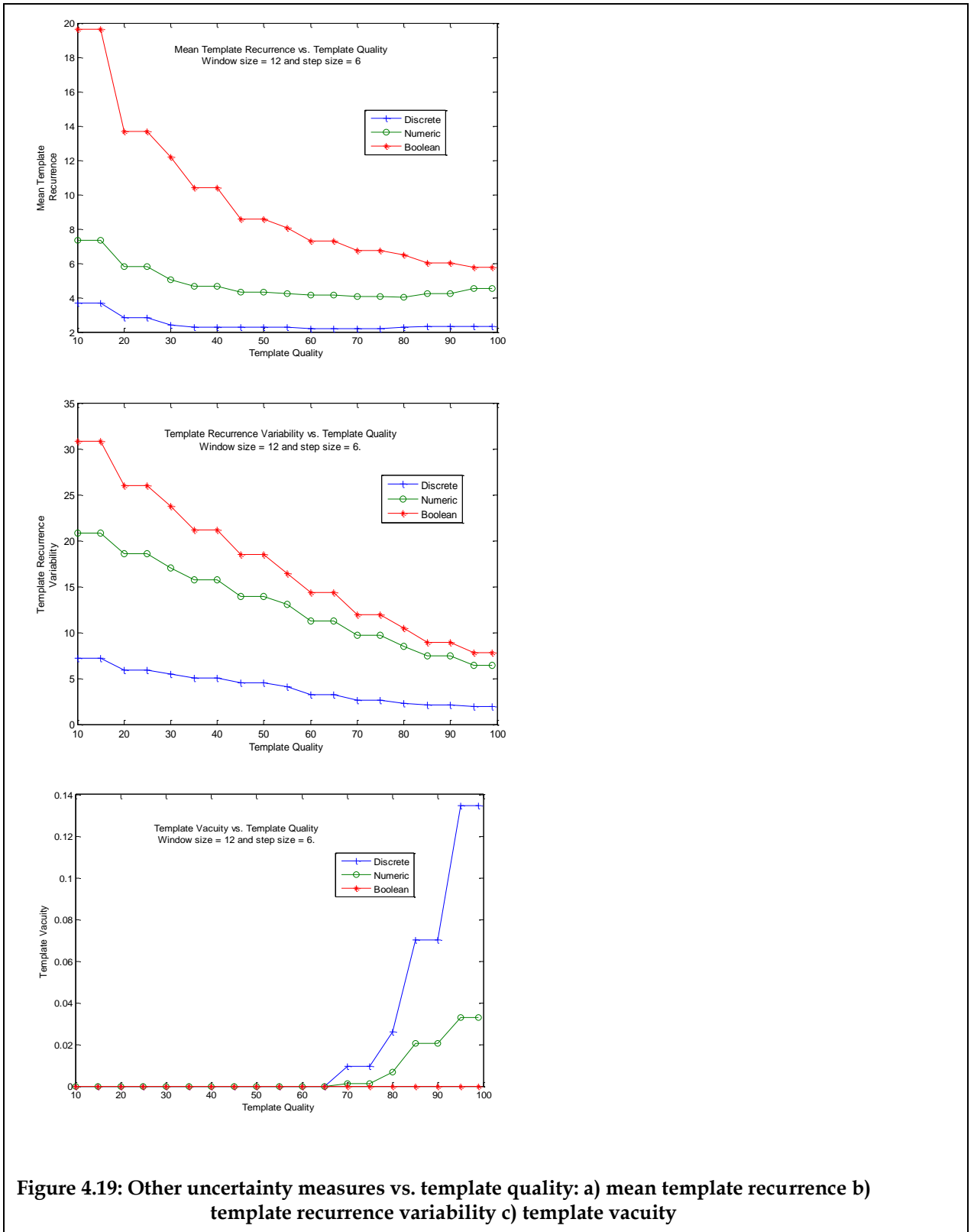
Figure 4.19: Other uncertainty measures vs. template quality: a) mean template recurrence b) template recurrence variability c) template vacuity

Figure 4.19a shows that the mean template recurrence decreases with the increase of the template quality for all datasets, but an abrupt decrease is observed for Boolean dataset. The general trend of the decrease in mean template recurrence can be justified by the fewer number of high quality wider templates, while the sharp fall of the template recurrence for Boolean dataset may be due to the fewer templates in Boolean datasets compared to other datasets. The higher the value of mean template recurrence, the lower the value of uncertainty in information systems.

Figure 4.19b shows that the mean template recurrence variability decreases with the increase of the template quality for all datasets but an abrupt decrease is observed for the Boolean dataset. The general trend of the decrease in mean template recurrence variability can be justified by the fewer number of high quality wider templates, while the sharp fall of the template recurrence variability for the Boolean dataset may be due to the fewer templates in Boolean datasets compared to other datasets. The lower value of template recurrence variability is desirable in the context of uncertainty management.

Fig 4.19c demonstrates the sensitivity of the template vacuity with the template quality. The increase in the template vacuity with the increase of template quality for discrete and numeric datasets can result from the higher number of vacuous templates that cannot satisfy the additional quality requirements. However, the

result shows that the template quality does not have any significant impact on the template vacuity for the Boolean dataset. This may be due to the fewer number of templates in the Boolean dataset.

4.5.2.8 The Correlation Matrix and the Statistical Significance

Table 4. 18: The correlation matrix and the statistical significance in the context of the window size (W = 6-78 hrs, Q = 70%)

	Window Size	Unique Template	Mean Template Recurrence	Mean Template Recurrence Variance	Entropy	Template Vacuity	Non-specificity
Window Size		–	–	–	–	+	+
Unique Template	–		+	+	+	–	–
Mean Template Recurrence	–	+		+	+	–	–
Mean Template Recurrence Variance	–	+	+		+	–	–
Entropy	–	+	+	+			–
Template Vacuity	+	–	–	–	–		+
Nonspecificity	+	–	–	–	–	+	

The correlation matrix signifies the correlation among the uncertainty measures. The correlation coefficient matrix (R) is calculated from the input matrix (X) where the rows symbolize the observations and the columns characterize the uncertainty measures. Each element of correlation coefficient matrix (R) is computed, as explained in Chapter 3, by employing the following formula:

$R(i,j) = \frac{C(i,j)}{\sqrt{C(i,i)C(j,j)}}$ where, the covariance matrix C = cov(X); i and j symbolize the

row and the column of the matrix R. After computing the correlation coefficient matrix, we generate another matrix, called matrix (P), to determine the statistical significance of the correlation among the uncertainty measures and to test the null hypothesis that there is no correlation among the uncertainty measures.

Table 4.19: The correlation matrix and the statistical significance in the context of template quality (W = 12 hrs, Q = 10-90%)

	Quality	Unique Template	Mean Template Recurrence	Mean Template Recurrence Variance	$H(T)/H(T)_{max}$	Template Vacuity	Non-specificity
Quality		−	−	−	+	+	−
Unique Template	−		+	+	−	−	+
Mean Template Recurrence	−	+		+	−	NS	+
Mean Template Recurrence Variance	−	+	+		−	−	+
$HT)/H(T)_{max}$	+	−	−	−		NS	−
Template Vacuity	+	−	NS	−	NS		−
Non-specificity	−	+	+	+	−	−	

Tables 4.18 and 4.19 show the statistical significances of the correlation of several uncertainty measures in the context of variable window size and variable quality, respectively. For a pair of uncertainty measures, such as the entropy and the unique templates, the p-value < 0.05 and the correlation coefficient $r > 0.5$ implies that the pairs are positively correlated in 95% confidence level. Similarly, the pair

of nonspecificity and unique template where $p < 0.05$ and $r < 0.5$ implies that they are negatively correlated at the 95% confidence level. The correlation between the pair of template vacuity and normalized entropy is not statistically significant because the p-value > 0.05. In general, '+' indicates positive correlation, '_' signifies negative correlation, and 'NS' implies non-significant correlation between a pair of measures.

The confidence bounds are computed based on the assumptions that the sample size is large and X has a multivariate normal distribution. Fig 4.20 shows that the empirical dataset has normal distribution.

Fig 4.20 Comparison of sensor data distribution with normal distribution

Even when the assumptions do not hold, we can use this technique to determine the statistical significance of correlation coefficients by employing bootstrap

sampling and generating a large sample when X has a multivariate normal distribution.

Once we know the correlation between all possible pairs of uncertainty measures and their statistical significance of the correlations in 95% confidence level, we can uncover the tradeoff among the uncertainty measures. The knowledge of the tradeoff is very important for developing an uncertainty-based optimization model.

CHAPTER V
RESEARCH CONCLUSIONS

5.1 Introduction

This research is motivated by the limited support of uncertainty management in the growing and promising technology of WSN. We found the research problem using an extensive literature review in the context of data aggregation and the uncertainty management in WSN. The principle contribution of this thesis is to provide a formalism for pattern-based data aggregation and an uncertainty management scheme based on rough set theory. The results are supported by validating the RS rules and establishing the statistical significance of the correlations among the uncertainty measures in a 95% confidence level. This chapter reviews the research results, summarizes the research conclusions, and ends with a list of future directions.

5.2 Research Conclusion and Summary

Wireless sensor networks are a growing research area, with the applications in wide range of fields including environmental monitoring, health monitoring,

surveillance, security, and so forth. These networks have the potential to observe and understand large-scale real-world phenomena at a fine spatio-temporal resolution and help us with critical decision making. However, severe resource constraints, frequent changes in the dynamics of the environment, and large numbers of unattended sensors pose uncertainties and limit their potential use in real life applications.

The uncertainties in sensor data streams may stem from missing data or unreliable data. Missing data may arise during sensor reading, format conversion, data discretization, data aggregation, data routing, data savings to storage devices, incorrect data labeling, etc. Unreliable data may result from random noise, actuator uncertainty, sensor status uncertainty, limited sensing ranges, compromised nodes, improper channels, transmission collisions, routing uncertainty, and resource uncertainty (Y. Liu & Das, 2006).

The problem of characterizing uncertainty in complex systems, such as sensor networks, is inherently interdisciplinary and it is difficult to completely capture the nature of uncertainty and cover all its aspects because of its complex nature and its propagation through all epistemological levels of a system by varying degrees (Ayyub & Klir, 2006). Some initiation is made in the area of location or deployment uncertainty of sensors, but, from the literature reviewed in Chapter

2 of the dissertation, it is evident that the uncertainty issues in wireless sensor networks is largely unexplored.

Given the growing demands for complex domain specific applications of sensor networks such as object tracking, pattern generation, and event identification, it is imperative to deal with uncertainty and uncertainty propagation through all the epistemological levels of a system in a real time environment. Another important issue in WSN is in-network data aggregation which saves energy, a major resource constraint in battery operated wireless sensor networks. A data aggregation technique that exploits the spatial and temporal correlation in sensor data at the node level has a potential to significantly reduce the data communication to the sink (Kargupta, 2007). The feature selection and reduction are also critical for large-scale attribute-oriented WSN.

In Chapter 2, we explained numerous types of uncertainties and several well-established mathematical formalisms to quantify and manipulate these types of uncertainties. We also reinvestigated and readdressed these uncertainties and the existing formalisms for characterizing theses uncertainties in the context of WSN. All of the uncertainty formalisms were developed and investigated in several application domains. Each of them is capable of handling certain types of uncertainties. In general, probability theory is appropriate for handling uncertainties caused by random components. Fuzzy set theory is well-known for

managing vague concepts (e.g. linguistic attributes). Dempster-Shafer evidence theory can be used to quantify the uncertainty due to information incompleteness. Rough set theory can be employed to characterize and quantify the uncertainty when it stems from coarseness. However, there is no unified mathematical formalism that integrates all existing formalisms and addresses real-life uncertainties in wireless sensor networks that are often a combination of several type of uncertainties (Nguyen et al., 2007).

We also reviewed several formalisms for data aggregation and rule generation, and investigated their possibilities in the context of WSN. Rough set theory appears to be the appropriate formalism for data aggregation as well as rule generation in WSN because of its well-known feature reduction capability. Other advantages of RST in the context of WSN include the capability of handling numeric as well as symbolic data, preservation of data semantics, shorter training time, ease of rule extraction, simple rule comprehensibility, data-driven approach, and independency on external parameters.

In Chapter 3, we presented the theoretical foundation of spatio-temporal pattern and rule generations and the uncertainty management in the pattern generation. The formalism for pattern-based data aggregation is explained in the context of WSN. A quality threshold is used to determine the interesting patterns. After discovering the patterns from the data, the next step is rule generation. In order

to reduce data communication from the cluster head to the sink, an important step is to extract and eliminate the redundant patterns which are not useful for decision making before generating the rules from the multi-valued decision system. The association rules are generated from the reducts, the minimal set of attributes to preserve the equivalence relation. A rough set based heuristic algorithm is used to determine the reducts by employing the degree of dependency as a measure.

The mathematical foundation of the hybrid model based on the rough set theory and a pattern-based data aggregation method is established in Chapter 3. In this chapter we also provided the definition of several uncertainties that are identified in the context of pattern generation in WSN. We also provided the mathematical representation of each uncertainty measure and the research methodology to quantify and manipulate these uncertainties.

In Chapter 4, we provided the validation of a mathematical formalism for uncertainty management in wireless sensor networks and the validation for the rules generated by rough set based spatio-temporal pattern discovery scheme. We employed the formalism in real world sensor data to find the homogeneous patterns in sea surface temperature (SST) and to generate the association rules. The formalism is tested on the discrete, numeric, and Boolean datasets. For each dataset, the experiment was repeated for several template qualities and window

sizes and the results were summarized in Chapter 4. Table 4.17 shows that 222, 264, and 317 temporal rules are generated from Boolean, numeric, and discrete datasets, each with 4368 observations, which implies that a significant data reduction can be achieved by employing RS rule mining on sensor data. The result also shows a fewer number of templates and association rules for Boolean datasets, compared to the other datasets, which implies a larger reduction in data communications in Boolean datasets. The results are illustrated in tables as well as in graphs. The result is consistent with the theory developed in Chapter 3.

Given that the rules generated at the cluster heads are sent periodically to the sink instead of the datasets, it is important to ensure that the rules faithfully represent the datasets. Thus we validate the rules by constructing a confusion matrix where each column denotes the predicted templates, each row represents the actual templates, the diagonal elements correspond to the correctly classified templates, and the off-diagonal elements signify misclassified templates. The confusion matrix is created from several parts of the patterns to confirm the self consistency of the rules (Bologna, 2002a, 2002b; Craven & Shavlik, 1999) and the accuracy of classification is computed. The results show an average accuracy of 89% for the template quality of 90%.

The uncertainties in the pattern generation of sea surface temperature (SST) stemming from imprecise data or missing data from sensors are characterized

and quantified by employing a rough set and Dempster-Shafer evidence theories. Finally, the correlations among the uncertainty measures are identified and the statistical significances of their correlations are established. The proposed model was tested on the discrete, numeric, and Boolean datasets. For each dataset, the experiment was repeated for several template qualities and window sizes and the results were summarized in several graphs. The graphs show the tradeoff among every possible uncertainty measures identified in the context of pattern generation for all three datasets. This result is consistent with the theory developed in Chapter 3. Finally, the correlations among the uncertainty measures were established in a 95% confidence level.

In summary, in this research we identified the useful and interesting spatio-temporal patterns from imprecise and uncertain sensor datasets by employing a rough-set rule induction method and provided a hybrid model of rough set theory and pattern-based data aggregation formalism to characterize and quantify the uncertainties in the context of pattern generation in WSN. The RS based feature selection plays an important role while generating the spatio-temporal patterns by removing the redundant features that are irrelevant for decision making.

The proposed research has the potential to produce significant contributions in the area of sensor data mining, data aggregation, intelligent feature selection,

data streams processing, knowledge discovery, and uncertainty management in wireless sensor networks. In particular, the specific contributions are as follows:

- We provide a formalization of a novel pattern-based data aggregation technique that has a potential to reduce data communication since only the rules are sent to the sink. The formalism discovers the spatio-temporal patterns and generates RS rules from the sensor data stream by intelligent feature selection. It not only provides the support for the symbolic and quantitative data but also for the outlier data. We provide the foundation of the uncertainty management in real time sensor database systems for continuous data.

- We develop a framework for the characterization and the quantification of uncertainties in wireless sensor networks environments by employing a hybrid model of uncertainty management based on rough set theory and pattern-based data aggregation formalism. We also demonstrate the trade off among the uncertainty measures in the context of WSN.

- We present a data-driven approach which is inherently non-invasive in nature. It also preserves the underlying data semantics. The proposed scheme does not require a normal distribution of the dataset.

5.3 Future Directions

The current research can be used as a framework to open up new directions in research. The challenges in this area of research and the possible future works can be summarized as follows:

- Development of the aggregation-driven routing protocols for wireless sensor networks where energy, bandwidth, power, networks lifetime, security as well as uncertainty are considered.

- Design of an optimization model based on uncertainty formalism beyond classical information theory in WSN. The disadvantage of the optimization based on the principle of maximum entropy is that it may lead to nonlinear programming (Klir and Yuan 1995). On the other hand, the optimization based on the nonspecificity lead to linear programming. Other types of uncertainties can be used if the non-specificity based optimization cannot provide a unique solution.

- Development of a multi-objective Pareto optimization model where identifying the Pareto Front from a set of points in a multi-objective space is the most important task. Multi-objective space in the context of WSN may include energy, bandwidth, power, security, and uncertainty. It is important to represent the possible trade-offs among the multiple conflicting objectives. The common approach to solve the multi-objective optimization is combining multiple objectives into a parametric scalar objective by using a weighted sum function (Huang, Fery, Xue, & Wang, 2008). Pareto optimization ensures that no further Pareto improvement is possible beyond the Pareto Front (i.e. it is not possible to improve any of the objectives without deteriorating the others) (Sushmita Mitra & Banka, 2006).

- Development of a real time concurrent environment that manages sensor information dynamically by transforming rough set rules to Petri Nets. The execution of Petri nets is nondeterministic and multiple transitions can be enabled at the same time in the Petri nets. Since Petri nets can have multiple tokens and one of their transitions may fire, they are well suited for modeling concurrent behavior of distributed systems, such as WSN (Skowron & Suraj, 1995).

- Development of a data aggregation scheme where rules are generated at the cluster heads from the temporal information system using dominance based rough set theory. In DRST, each attribute of the temporal information system is assigned a cost or gain function with respect to the decision attribute, and the attributes that are not directly associated with cost or gain function are labeled as none. By generating a dominance matrix and performing approximations of *upward unions* and *downward unions* of uncertainty classes, one can determine the reducts or the optimal set of attributes indespensible for decision making. The advantages of using DRST include that it does not require discretization for numerical attributes, nor does it require the removal of missing values (Iftikhar U. Sikder, Mal-Sarkar, & Mal, 2006).

- Design of a rule-based automated or semi-automated expert system by automatically inducing if-then rules from the empirical data. Automated expert systems can be very useful for critical decision making in real time where human access is infeasible.

- Expansion of the model where several security issues are addressed. The proposed scheme can easily incorporate security measures by encrypting the templates or patterns before sending to the sinks. This approach has the potential to reduce the cost associated with the security management since only the templates are required to be encrypted and decrypted, instead of encrypting and decrypting the entire temporal information system. With the increase of the network density and data correlation, this reduction in the cost can be significant.

However, the basic framework that has been proposed in this dissertation may require significant modifications of boundary conditions and assumptions to accommodate the future directions outlined above. Some of the domain constraints of rough set theory may require further evaluation.

BIBLIOGRAPHY

Akkaya, K., & Younis, M. (2005). A Survey on Routing Protocols for Wireless Sensor Networks. *Ad Hoc Networks, 3*, 325-349.

Akyildiz, I. F. (2005). *Underwater Acoustic Sensor Networks: Research Challenges*. Paper presented at the In the Tyrrhenian International Workshop on Digital Communications.

Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). Wireless Sensor Networks: A Survey. *Computer Networks, 38*, 393-422.

Al-Qaheri, H., Hassanien, A. E., & Abraham, A. (2008). Discovering Stock Price Prediction Rules using Rough Sets. *Neural Network World 18*(3), 181-198.

Ayyub, B. (2003). *Risk Analysis in Engineering and Economics*: Chapman and Hall/CRC Press.

Ayyub, B., & Klir, G. (2006). *Uncertainty Modeling and Analysis in Engineering and the Sciences* (1st ed. Vol. 1). New York: Chapman & Hall/CRC Taylor & Francis Group.

Ayyub, B., & McCuen, R. (2003). *Probability, Statistics, and Reliability for Engineers and Scientists*. Boca Raton, FL: Chapman & Hall/CRC Press.

Babcock, B., Babu, S., Datar, M., Motwani, R., & Widom, J. (2002). *Models and Issues in Data Stream Systems.* Paper presented at the In the Proceedings of the 21nd Sumposium on Principles of Database Systems, New York, USA.

Baggio, A. (2005, June, 2005). *Wireless Sensor Networks in Precision Agriculture.* Paper presented at the In the Proceedings of the Workshop on Real-World Wireless Sensor Networks (REALWSN 2005), Stockholm, Sweden.

Banerjee, M., Mitra, S., & Pal, S. K. (1998). Rough Fuzzy MLP: Knowledge Encoding and Classification. *IEEE Transactions on Neural Networks, 9*(6), 1203-1216.

Bazan, J. (1996, July 1-5). *Dynamic Reducts and Statistical Inference.* Paper presented at the Proceedings of Information Processing and Management of Uncertainty on Knowledge Based Systems (IPMU96), Granada, Spain, Universidad de Granada.

Bazan, J., Skowron, A., & Synak, P. (1994). *Discovery of Decision Rules from Experimental Data.* Paper presented at the Proceedings of the Third International Workshop on Rough Sets and Soft Computing, San Jose, California.

Bellman, R. (1961). *Adaptive Control Processes*. Princeton: Princeton University Press.

Beynon, M., & Peel, M. (2001). Vaiable Precision Rough Set Theory and Data Discrimination: An Application to Corporate Failure Prediction
OMEGA, 29, 561-576.

Bologna, G. (2002a). On the Validation of the DIMLP Neural Network. *In Proceedings of FLAIRS Conference*.

Bologna, G. (2002b). Rule Extraction from Bagged Neural Networks. *In Proceedings of Hybrid Intelligent Systems*.

Braginsky, D., & Estrin, D. (2002, October, 2002). *Rumor Routing Algorithm for Sensor Networks.* Paper presented at the in Proceedings of the First Workshop on Sensor Networks and Applications (WSNA), Atlanta, GA.

Buttyán, L., Schaffer, P., & Vajda, I. (2006, October 30, 2006). *RANBAR: RANSAC-based Resilient Aggregation in Sensor Networks.* Paper presented at the Paper presented at the Workshop on Security of Ad Hoc and Sensor Networks, Proceedings of the Fourth ACM Workshop on Security of Ad Hoc and Sensor Networks, Alexandria, VA.

Cam, H., Özdemir, S., Nair, P., Muthuavinashiappan, D., & Sanli, H. O. (2006). *Computer Communications, 29*, 446-455.

Cantoni, V., Lombardi, L., & Lombardi, P. (2006). Challenges for Data Mining in Distributed Sensor Networks. *ICPR, 1*, 1000-1007.

Cattani, C., & Kudreyko, A. (2008). On the Discrete Harmonic Wvelet Transform. *Mathematical Problems in Engineering, 2008*, 1-7.

Chakrabarti, K., Garofalakis, M. N., Rastogi, R., & Shim, K. (2001). Approximate query processing using wavelets. *VLDB J. , 10*(2-3), 199-223.

Chang, J., & Lee, W. (2003). *Finding recent frequent itemsets adaptively over online data streams.* Paper presented at the Prooceedings of the ACM SIGKDD.

Chatterjee, M., and, S. K. D., & Turgut, D. (2002). WCA: A Weighted Clustering Algorithm for Mobile Ad hoc Networks. *Journal of Cluster Computing (Special Issue on Mobile Ad hocNetworks), 5*(2), 193–204.

Chatterjee, M., Das, S. K., & Turgut, D. (2002). WCA: A Weighted Clustering Algorithm for Mobile Ad hoc Networks. *Journal of Cluster Computing, 5*, 193-204.

Cheng, R., & Prabhakar, S. (2003). Managing Uncertainty in Sensor Databases. *ACM SIGMOD Record, 32*(4), 41-46.

Chouchoulas, A. (2001). Incremental Feature Selection based on Rough Set Theory [Electronic Version]. *PhD Proposal*, 50. Retrieved August 5, 2009 from www.cisa.inf.ed.ac.uk/proposals/AlexiosChouchoulas.ps.

Chu, M., Haussecker, H., & Zhao, F. (2002). Scalable Information-Driven Sensor Querying and Routing for Ad Hoc Heterogenous Sensor Networks. *The International Journal of High Performance Computing Applications, 16*(3).

Cormode, G., & Muthukrishnan, S. (2005). What's Hot and What's Not: Tracking Most Frequent Items Dynamically. *ACM Transactions on Database Systems (TODS), 30*(1), 249 - 278

Craven, M., & Shavlik, J. (1999). *Rule Extraction: Where Do We Go from Here?* : University of Wisconsin Machine Learning Research Group Working Paper 99-1.

Crossbow Technology Inc. (Publication (2005). Retrieved 4 Sept 2008: http://www.xbow.com/Industry_solutions/EnvironmentalMonitoring.aspx

Daubie, M., Levecq, P., & Meskens, N. (2002). A Comparison of Rough Sets and Recursive Partitioning induction Approaches: An Application to Commercial Loans. *International Transactions in Operation Research, 9*, 681-694.

Dempster, A. P. (1967). Upper and Lower Probabilities Induced by a Multi-valued Mapping. *Annals of Mathematical Statistics, 38*, 325-339.

Dempster, A. P. (1968). A Generalisation of Bayesian Inference (With Discussion). *Journal of Royal Statistical Society B, 30*, 205-232.

Devijver, P., & Kittler, J. (1982). *Pattern Recognition: A Statistical Approach*. Englewood Cliffs, NJ Prentice-Hall.

Dhar, V., & Tuzhilin, A. (1993). Abstract-Driven Pattern Discovery in Databases. *IEEE Transactions on Knowledge and Data Engineering, 5*(6), 926-937.

Ding, M., Cheng, X., & Xue, G. (2003, October, 2003). *Aggregation Tree Construction in Sensor Networks.* Paper presented at the Paper presented at the 2003 IEEE 58th Vehicular Technology Conference.

Dougherty, J., Kohavi, R., & Sahami, M. (1995). *Supervised and Unsupervised Discretization of Continuous Features.* Paper presented at the In Machine Learning:Proceedings of the 12th International Conference.

Dubois, D., & Prade, H. (1990). Rough Fuzzy Sets and Fuzzy Rough Sets. *International Journal of General Systems, 17*, 191-209.

Düntsch, I., & Gediga, G. (1998). Uncertainty Measures of Rough Set Prediction. *Artificial Intelligence 106*(1), 109-137.

Elmagarmid, A. K., Ghanem, T. M., Hammad, M. A., Mokbel, M. F., & Aref, W. G. (2007). Incremental Evaluation of Sliding-Window Queries over Data Streams. *IEEE Transations on knowledge and Data Engineering (TKDE), 19*(1), 57-72.

Elnahrawy, E. (2003). *Research Directions in Sensor Data Streams: Solutions and Challenges* (No. DCIS-TR-527): Rutgers University.

Elson, J., & Estrin, D. (2004). Wireless Sensor Networks: A Bridge to the Physical World [Electronic Version]. *Center for Embedded Network Sensing.*, 1-21. Retrieved January 1.

Fayyad, U. M., & Irani, K. B. (1993). *Multi-interval Discretization of Continuous-valued Attributes for Classification Learning.* Paper presented at the 13th Int. Joint Conference on AI, Chambery, France.

Fu, L. M. (1999). Knowledge Discovery based on Neural Networks. *Communications of the ACM, 42*(11), 47-50.

Gaber, M. M., Krishnaswamy, S., & Zaslavsky, A. (2005). On-board Mining of Data Streams in Sensor Networks In U. M. Sanghamitra Bandyopadhyay, Lawrence B. Holder and Diane J. Cook (Ed.), *Advanced Methods for Knowledge Discovery from Complex Data* (pp. 307-335). London: Springer Verlag.

Gaber, M. M., Zaslavsky, A., & Krishnaswamy, S. (2005). Mining Data Streams: A Review. *SIGMOD Record, 34*(2), 18-26.

Ganoulis, A. (2007). *Fuzzy Modelling for Uncertainty Propagation and Risk Quantification in Environmental Water Systems*: IOS Press.

Gibbons, P., & Tirthapura, S. (2002). *Distributed Streams Algorithms for Sliding Windows.* Paper presented at the In Proceedings of ACM Symposium on Parallel Algorithms and Architectures.

Golab, L., & Özsu, M. T. (2003). Issues in Data Stream Management. *SIGMOD Record, 32*(2), 5-32.

Greco, S., Matarazzo, B., & Slowinski, R. (2002). Rough Approximation by Dominance Relations. *International Journal of Intelligent Systems, 17*(2), 153-171.

Grzymala-Busse, J. W. (1991). *Managing Uncertainty in Expert Systems*. Dordrecht, Netherlands: Kluwer Academic Publishers.

Grzymala-Busse, J. W. (2003). *MLEM2 - Discretization During Rule Induction.* Paper presented at the Proceedings of the International Conference on Intelligent Information Processing and WEB Mining Systems (IIPWM'2003), Zakopane, Poland.

Grzymala-Busse, J. W. (2007). Mining Numerical Data—A Rough Set Approach In M. K. e. al. (Ed.), *Rough Sets and Intelligent Systems Paradigms* (Vol. 4585/2007, pp. 12-21). Berlin Springer-Verlag.

Guha, S., & Harb, B. (2005). *Wavelet Synopsis for Data Streams: Minimizing Non-euclidean Error.* Paper presented at the International Conference of Knowledge Discovery and Data Mining. , Chicago, USA.

Hefeeda, M. (2008). Forest Fire Modeling and Early Detection using Wireless Sensor Networks. *Ad Hoc & Sensor Wireless Networks, 7*, 169-224.

Heinzelman, W. (2000). *Application-specific Protocol Architectures for Wireless Networks.* Unpublished PhD Thesis, Massachusetts Institute of Technology.

Heinzelman, W., Chandrakasan, A., & Balakrishnan, H. (2000, January, 2000). *Energy-Efficient Communication Protocol for Wireless Microsensor Networks.* Paper presented at the in the Proceedings of 33rd Hawaii International Conference on System Sciences (HICSS '00), Hawaii.

Heinzelman, W., Kulik, J., & Balakrishnan, H. (1999, August, 1999). *Adaptive Protocols for Information Dissemination in Wireless Sensor Networks.* Paper presented at the in the Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99), Seattle, WA.

Heinzelman, W. R., Chandrakasan, A. P., & Balakrishnan, H. (2002). An Application-specific Protocol Architecture for Wireless Microsensor Networks. *IEEE Transactions Wireless Communications*, 660-670.

Hopfield, J. (1982). Neural Networks and Physical Systems with Emergent Collective Computational Abilities. *Proceedings of the National Academy of Sciences, 79*(8), 2554-2558.

Howard, A., Matari´c, M. J., & Sukhatme, G. (2001, October, 2001). *Relaxation on a Mesh: a Formalism for Generalized Localization.* Paper presented at the Paper presented at the Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Wailea, Hawaii.

Huang, B., Fery, P., Xue, L., & Wang, Y. (2008). Seeking the Pareto Front for Multiobjective Spatial Optimization Problems. *International Journal of Geophraphical Information Science, 22*(5), 507-526.

Intanagonwiwat, C., Govindan, R., & Estrin, D. (2000, August, 2000). *Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks.* Paper presented at the Paper presented at the Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCOM'00), Boston, MA.

Intel Research (Publication (2005). Retrieved Sep. 4, 2008:
www.intel.com/research/exploratory/wireless_sensors.htm

Intel Research (Publication (2008). Retrieved Sep. 4, 2008:
www.intel.com/research/exploratory/wireless_sensors.htm

Jensen, R., & Shen, Q. (2004). Fuzzy-rough Attribute Reduction with Application to Web Categorization. *Fuzzy Sets Syst., 141*, 469-485.

Jenssen, T.-K., Komorowski, H. J., & Øhrn, A. (1998). Some Heuristics for Default Knowledge Discovery. *Rough Sets and Current Trends in Computing*, 373-380.

Jiang, N., & Gruenwald, L. (2006). Research Issues in Data Stream Association Rule Mining. *SIGMOD Record, 35*(1), 14-19.

Jolliffe, I. T. (1986). *Principal Component Analysis*. New York: Springer-Verlag.

Kargupta, H. (2007). *Learning from Data Streams Processing Techniques in Sensor Networks*. New York: Springer.

Kargupta, H., Puttagunta, V., Klein, M., & Sarkar, K. (2006). On-board Vehicle Data Stream Monitoring Using MineFleet and Fast Resource Constrained Monitoring of Correlation Matrices. *New Generation Comput, 25*(1), 5-32.

Klir, G. J., & Folger, T. A. (1988). *Fuzzy Sets, Uncertainty and Information*. Englewood Cliffs, NJ: Prentice-Hall Inc.

Klir, G. J., & Wireman, M. J. (1999). *Uncertainty-based information: elements of generalized information theory*. Heidelberg and New York: Physica-verlag/Springer-Verlag.

Kolmogorov, A. N. (1956). *Foundations of the Theory of Probability* (N. Morrison, Trans.). Chelsea, New York: Springer-Verlag.

Kowalczyk, W., & Piasta, Z. (2006). Rough-set Inspired Approach to Knowledge Discovery in Business Databases. In S. B. Heidelberg (Ed.), *Research and Development in Knowledge Discovery and Data Mining* (Vol. 1394/1998, pp. 186-197): Springer Berlin / Heidelberg.

Krishnamachari, B. (2005). *Networking Wireless Sensors*. New York: Cambridge University Press.

Krishnamachari, B., Estrin, D., & Wicker, S. (2002). *The Impact of Data Aggregation in Wireless Sensor Networks.* Paper presented at the In the Proc. 22nd International Conference on Distributed Computing Systems Workshops.

Krusinska, E., Slowinski, R., & Stefanowski, J. (1992). Discriminant Versus Rough Sets Approach to Vague Data Analysis. *Journal of Applied Stochastic Models and Data Analysis, 8*.

Lakshmanan, L., Leone, N., Ross, R., & Subramanian, V. (1997). ProbView: A Flexible Probabilistic Database System. *ACM Transactions on Database Systems, 22*, 419-469.

Laxman, S., Sastry, P., & Unnikrishnan, K. (2007). Discovering Frequent Generalized Episodes When Events Persist for Different Durations. *IEEE Transactions on Knowledge and Data Engineering 19*(9), 1188-1201.

Li, Z., Nadon, S., & Cihlar, J. (2000). Satellite-based detection of Canadian boreal Forest Fires: Development and Application of the Algorithm. *International Journal of Remote Sensing, 21*, 3057-3069.

Lindley, D. V. (1975). *Making Decisions*. Chichester, UK: John Wiley & Sons

Lindsey, S., & Raghavendra, C. (2002, March, 2002). *PEGASIS: Power Efficient GAthering in Sensor Information Systems.* Paper presented at the in the Proceedings of the IEEE Aerospace Conference, Big Sky, Montana.

Lindsey, S., Raghavendra, C., & Sivalingam, K. M. (2002). Data Gathering Algorithms in Sensor Networks Using Energy Metrics. *IEEE Transactions on Parallel and Distributed Systems, 13*, 924-935.

Liu, P., El-Darzi, E., & Lei, L. (2008). Applying data mining algorithms to inpatient dataset with missing values. *Journal of Enterprise Information Management, 21*(1), 81-92.

Liu, Y., & Das, S. K. (2006). Information Intensive Wireless Sensor Networks: Potential and Challenges. *IEEE Communications 44* (11), 142–147.

Lohi, A., Ikola, T., Rauste, Y., & Kelha, V. (1999). *Forest Fire Detection with Satellites for Fire Control.* Paper presented at the In Proceedings of IUFRO Conference on Remote Sensing and Forest Monitoing Rogow, Poland.

Mak, B., & Munakata, T. (2002). Rule Extraction from Expert Heuristics: A Comparative Study of Rough sets with Neural Networks and ID3. *European Journal of operation research, 136*(1), 212-229.

Mal-Sarkar, S., Sikder, I. U., Yu, C., & Konangi, V. K. (2009). Uncertainty-aware Wireless Sensor Networks *International Journal of Mobile Communications 7*(3), 330-345.

Manjeshwar, A., & Agrawal, D. P. (2001, April, 2001). *TEEN: A Protocol for Enhanced Efficency in Wireless Sensor Networks.* Paper presented at the in the Proceedings of the 1st International Workshop on Parallel and Distributed Issues in Wireless Networks and Mobile Computing, San Francisco, CA.

Manjeshwar, A., & Agrawal, D. P. (2002). *APTEEN: A Hybrid Protocol for Effcient Routing and Comprehensive Information Retreval in Wireless Sensor Networks.*

Paper presented at the in the Proceedings of the 2nd International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, Fort Lauderdale, FL.

Mannila, H., Toivonen, H., & Verkamo, A. I. (1997). Discovery of Frequent Episodes in Event Sequences. *Data Mining and Knowledge Discovery, 1*(3), 259-289.

Mitra, S., & Acharya, T. (2003). *Data Mining: Multimedia, Soft Computing, Bioinformatics* (1 ed.). Hoboken, New Jersey: John Wiley & Sons, Inc.

Mitra, S., & Banka, H. (2006). Multi-objective Evolutionary Biclustering of Gene Expression Data. *Pattern Recognition, 32*(12), 2464-2477.

Mitra, S., Pal, S. K., & Mitra, P. (2002). Data Mining in Soft Computing Framework: A Survey. *IEEE Transactions on Neural Networks, 13*, 3-14.

Morrissey, J. M. (1990). Imprecise Information and Uncertainty in Information Systems. *ACM Transactions on Information Systems, 8*(2), 159-180.

Munakata, T. (2008). *Fundamentals of the New Artificial Intelligence: Neural, Evolutionary, Fuzzy and More* (2nd ed.): Springer.

Munakata, T., & Pawlak, Z. (1996). *Rough Control: Applications of Rough Set Theory to Control.* Paper presented at the Fourth European Congress on Intelligent Techniques and Soft Computing, Aachen, Germany.

NanoTechnology (Publication (2007). Retrieved Sep. 6, 2008: http://www.azonano.com/Details.asp?ArticleID=1318

Nguyen, H. T., Kreinvich, V., & Dhompongsa, S. (2007). Mathematical Foundation for Intelligent Technologies. *Information Sciences, 177*(16), 3215-3217.

Øhrn, A. (1999). Discernibility and Rough sets in Medicine: Tools and Applications. Unpublished PhD Thesis. Norweigian University of Science and Technology.

Pal, S. K., Shankar, B. U., & Mitra, P. (2005). Granular computing, rough entropy and object extraction. *Pattern Recognition Letters, 26*, 2509 - 2517.

Pal, S. K., & Skowron, A. (Eds.). (1999). *Rough–Fuzzy Hybridization: A New Trend in Decision Making*. Singapore: Springer.

Paluch, A., & Rzasa, W. (2005). Templates 1.0.   Retrieved 2007, from http://rsds.wsiz.rzeszow.pl

Pao, Y.-H. (1989). *Adaptive Pattern Recognition and Neural Networks* Addison-Wesley, Reading.

Patwari, N., & Hero, A. O. (2003). Relative Location Estimation in Wireless Sensor Networks. *IEEE Transactions On Signal Processing, 51*(8), 2137-2148.

Pawlak, Z. (1982). Rough Sets. *International Journal of Computer and Information Sciences, 11*(5), 341-356.

Pawlak, Z. (1992a). Preface. In R. Slowinski (Ed.), *Intelligent Decision Support: Handbook of Applications and Advances of the Rough Sets Theory*. Warsaw, poland: Kluwer the Language of Science.

Pawlak, Z. (1992b). *Rough Sets: Theoretical Aspects of Reasoning about Data*. Norwell, MA, USA: Kluwer Academic Publishers.

Pawlak, Z. (1997). Rough Set Approach to Knowledge-based Decision Support *European Journal of Operation Research, 99*, 48-57.

Pawlak, Z. (2004). Some Issues on Rough Sets. In *Transactions on Rough Sets I* (pp. 1-58). Berlin, Heidelberg: Springer.

Pawlak, Z., Grzymala-Busse, J., Slowinski, R., & Ziarko, W. (1995). Rough Sets. *Communications of the ACM, 38*(11), 89-95.

Pawlak, Z., Polkowski, L., & Skowron, A. (2005). Rough Sets. *Encyclopedia of Database Technologies and Applications*, 575-580.

Pawlak, Z., & Skowron, A. (1993). *A Rough Set Approach for Decision Rules Generation* (ICS Research Report 23/93 No. ICS Research Report 23/93): Warsaw University of Technology.

Pawlak, Z., & Slowinski, R. (1994). Rough Set Approach to Multi-attribute Decision Analysis. *European Journal of Operation Research, 72*, 443-459.

Peters, J. F., & Skowron, A. (2002). A Rough Set Approach to Knowledge Discovery. *International Journal of Intelligent Systems, 17*, 109-112.

Polkowski, L., & Skowron, A. (1998, June 22-26). *Rough Sets and Current Trends in Computing*. Paper presented at the RSCTC'98, Warsaw, Poland.

Polkowski, L., & Skowron, A. (1998). *Rough Sets in Knowledge Discovery 1 and 2*. Heidelberg: Physica-Verlag.

Quinlan, J. R. (1986). Induction of Decision Trees. *Machine Learning, 1*, 81-106.

Quinlan, J. R. (1992). *C4.5: Programs for Machine Learning* San Mateo, CA: Morgan Kaufman.

Rajagopalan, R., & Varshney, P. K. (2006). Data Gathering and Aggregation Techniques in Sensor Networks: A Survey. *IEEE Communications Surveys and Tutorials, 8*(1-4), 48-63.

Reznik, L., & Kreinovich, V. (2004). *Fuzzy and Probabilistic Models of Association Information in Sensor Networks.* Paper presented at the The 13th IEEE International Conference on Fuzzy Systems, Budapest, Hungary.

Rzasa, W., Paluch, A., & Suraj, Z. (2004). *Decision Rules in Multivalued Decision Systems.* Paper presented at the RSCTC 2004, LNAI 3066, Berlin, Heidelberg.

Sadagopan, N., Krishnamachari, B., & Helmy, A. (2003, May, 2003). *The ACQUIRE Mechanism for Efficient Querying in Sensor Networks.* Paper presented at the in the Proceedings of First International Workshop on Sensor Network Protocol and Applications, Anchorage, Alaska.

Safavian, S. R., & Landgrebe, D. (1991). A Survey of Decision Tree Classifier Methodology. *IEEE Transactions on Systems, Man, and Cybernetics, 21*, 660-674.

Schurgers, C., & Srivastava, M. B. (2001). *Energy efficient Routing in Wireless Sensor Networks.* Paper presented at the in the MILCOM Proceedings on Communications for Network-Centric Operations: Creating the Information Force, McLean, VA.

Shafer, G. (1976). *A Mthematical Theory of Evidence*. New Jersey: Princeton University Press.

Shah, R., & Rabaey, J. (2002, March, 2002). *Energy Aware Routing for Low Energy Ad Hoc Sensor Networks.* Paper presented at the in the Proceedings of the IEEE Wireless Comminications and Networking Conference (WCNC) Orlando, FL.

Shen, Q., & Chouchoulas, A. (2000). A Modular Approach to Generating Fuzzy Rules with Reduced Attributes for the Monitoring of Complex Systems. *Engineering Applications of Artificial Intelligence, 13*, 263-278.

Sikder, I. U. (2003). *Uncertainty Management in Empirical Knowledge Discovery and Classification in Spatial Databases.* Unpublished Doctoral Dissertation, University of Maryland, Baltimore.

Sikder, I. U., Mal-Sarkar, S., & Mal, T. K. (2006). Knowledge-Based Risk Assessment Under Uncertainty for Species Invasion *Risk Analysis, 26*(1), 239-252.

Sikder, I. U., & Munakata, T. (2009). Application of Rough Set and Decision Tree for Characterization of Premonitory Factors of Low Seismic Activity. *Expert System with Applications, 36*, 102-110.

Skowron, A. (1995). Extracting Laws from Decision Tables - A Rough Set Approach. *Computational Intelligence, 11*, 371-388.

Skowron, A., & Grzymala-Busse, J. (1994). From Rough Set Theory to Evidence Theory. In M. Fedrizzi, J. Kacprzyk & R. R. Yager (Eds.), *Advances in Dempster-Shafer Theory* (pp. 193-236). New York: John Wiley.

Skowron, A., & Polkowski, L. (1998). Rough Mereological Foundations for Design, Analysis, Synthesis, and Control in Distributive Systems. *Information Sciences, 104*(1-2), 129-156.

Skowron, A., & Suraj, Z. (1995). *Discovery of Concurrent Data Models from Experimental Tables: A Rough Set Approach* Warsaw Institute of Technology.

Slowinski, R. (Ed.). (1992). *Intelligent Decision Support*. Dordrecht: Kluwer Academic.

Solis, I., & Obraczka, K. (2003). *In-Network Aggregation Trade-offs for Data Collection in Wireless Sensor Networks* (No. Technical Report No. INRG 102). Santa Cruz: University of California Santa Cruz.

Stefanowski, J. (1992). Rough Sets Theory and Discriminant Methods as Tools for Analysis if Information Systems - A Comparative Study. *Foundations of Computing and Decision Science, 2*, 81-98.

Swiniarski, R. W., & Skowron, A. (2003). Rough Set Methods in Feature Selection and Recognition. *Pattern Recognition Letters, 24*(6), 833-849.

Synak, P. (2001). *Temporal Templates and Analysis of Time Related Data.* Paper presented at the RSCTC, LNAI 2005, Berlin, Heidelberg.

Synak, P. (2003). *Temporal Feature Extraction from Temporal Information Systems.* Paper presented at the ISMIS 2003, LNAI 2871, Berlin, Heidelberg.

Thangavel, K., & Pethalakshmi, A. (2009). Dimensionality Reduction Based on Rough Set Theory: A Review. *Applied Soft Computing, 9*, 1-12.

Tsumoto, S. (1999). *Discovery of Rules for Medical Expert Systems - Rough Set Approach.* Paper presented at the Third International Conference on Computational Intelligence and Multimedia Applications.

Ulrych, J. (2008, 2008). *Processing XML Data Streams: A Survey.* Paper presented at the In WD'08 Proceedings of Contributed Papers.

Vitter, J. S. (1985). Random Sampling with a Reservoir. *ACM Transactions on Mathematical Software, 11*(1), 37-57.

Wang, H., Yip, L., Yao, K., & Estrain, D. (2004, May 17-21, 2004). *Lower Bounds of Localization Uncertainty in Sensor Networks.* Paper presented at the Paper

presented at the IEEE International Conference on Acoustics, Speech, and Signal Processing, Quebec, Canada.

Wroblewski, J. (1995, September 28–October 1). *Finding Minimal Reducts using Genetic Algorithms.* Paper presented at the Proc. Second Annual Join Conf. on Information Sciences, Wrightsville Beach, NC.

Wu, D., & Mendel, J. M. (2007). Uncertainty Measures for Interval type-2 Fuzzy Sets. *Information Sciences, 177* (23), 5378-5393

Yager, R. R. (2000). Hierarchical Aggregation Functions Generated from Belief Structures. *IEEE Transactions on Fuzzy Systems, 8*, 481-490.

Yao, Y., & Gehrke, J. (2002). The Cougar Approach to In-network Query Processing in Sensor Networks. *SIGMOD Record.*

Zadeh, L. A. (1965). Fuzzy Sets. *Information and Control, 8*, 338-353.

Zadeh, L. A. (1971). Similarity Relations and Fuzzy Orderings. *Information Sciences, 3*, 177-200.

Zadeh, L. A. (1975). The Concept of a Linguistic Variable and its Application to Approximate Reasoning: Part 1, 2, and 3. *Information Sciences, 8, 8, 9*, 199-249, 301-357, 143-180.

Zadeh, L. A. (1978). Fuzzy Sets As a Basis for a Theory of Possibility. *Fuzzy Sets and Systems, 1*, 3-28, 23-28.

Zhang, W., Das, S. K., & Liu, Y. (2006, September 28, 2006). *A Trust based Framework for Secure Data Aggregation in Wireless Sensor Networks.* Paper presented at the in 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks, Reston, VA,.

Zhang, Y., & Edgar, T. F. (2006). *Bio-reactor Monitoring with Multiway-PCA and Model Based- PCA*. Paper presented at the AIChE Annual Meeting, San Francisco, CA.

Zhou, J., & Mourelatos, Z. P. (2008). *Design under Uncertainty using a Combination of Evidence Theory and a Bayesian Approach*. Paper presented at the 3rd International Workshop on Reliable Engineering Computing (REC 2008).

Zou, Y., & Chakrabarty, K. (2004). Uncertainty-aware and coverage-oriented deployment for sensor networks. *Journal of Parallel and Distributed Computing, 64*(7).