**Ministère de l'enseignement Supérieur et de la recherche Scientifique**
وزارة التعليم العالي والبحث العلمي

**Badji Mokhtar Annaba University**
**Université Badji Mokhtar – Annaba**
**Faculté de Technologie**

**جامعة باجي مختار – عنابـــــــــة**
**كـــــليـــــة التكـنـــولـــوجيـــا**

**Département Electronique**

**قســـــــم الإلكترونيك**

# Thesis

## Doctorate Third Cycle

**Speciality : Automatic and Signals**

By :
**BOUZID Soumia**

Entitled :

# Diagnostic et Contrôle des Systèmes de Distribution de l'eau Potable

Member of jury:

| N° | Name and Surname | Grade | Institution | Quality |
|---|---|---|---|---|
| **01** | DEBBACHE Nasr Eddine | Prof. | University of Annaba | President |
| 02 | CHENIKHER Salah | Prof. | University of Tebessa | Supervisor |
| 03 | RAMDANI Messaoud | Prof. | University of Annaba | Co-Supervisor |
| 04 | KHELIL Khaled | Prof. | University of Souk Ahras | Examiner |
| 05 | GHERBI Sofiane | Prof. | University of Annaba | Examiner |

2023-2024

# " Diagnostic et Contrôle des Systèmes de Distribution de l'eau Potable "


**الملخص:**

يعتمد نجاح أي استراتيجية تشخيص بشكل حاسم على أجهزة الاستشعار التي تقيس متغيرات العملية.
تقدم هذه الأطروحة طريقة أعطال مستشعر الكشف والتشخيص بناءً على شبكة عنق الزجاجة العصبية. يستخدم نهج شبكة عنق الزجاجة العصبية كأداة تحكم في العمليات الإحصائية لأنظمة توزيع مياه الشرب لاكتشاف وعزل أعطال أجهزة الاستشعار. يتم التحقق من صحة هذه الطريقة في المحاكاة على نظام غير خطي: خزان رباعي ونظام توزيع مياه الشرب الفعلي. الهدف من هذا البحث هو المساهمة في الإشراف على أنظمة توزيع مياه الشرب باستخدام طريقة تشخيص الشبكة العصبية. مع الطلب الكبير على إمدادات المياه بسبب النمو السكاني، فإن الموافقة على التكنولوجيا الجديدة لضمان جودة المياه بتكلفة أقل أمر ضروري.
تقدم هذه الأطروحة أنظمة توزيع مياه الشرب بناءً على تقنية نمذجة غامضة غير خطية. يستخدم النهج نموذجًا غامضًا متعدد المدخلات متعدد المخرجات تاكاجي سوجينو ، وهو مناسب لبناء فئة كبيرة من العمليات غير الخطية. تم التحقق من صحة الإطار المقترح على نظام توزيع مياه الشرب الحقيقي، وتم تنفيذ نموذج تاكاجي سوجينو الضبابي متعدد المدخلات متعدد المخرجات، في سياق الرقابة التنبؤية غير الخطية لتنظيم جودة المياه (تركيز الكلور في مياه الشرب) .
والهدف هو إبقاء نواتج النظام ضمن الحدود العليا والأدنى من متطلبات اللوائح الصحية. مراقبة التغذية المرتدة عبر الإنترنت لمخلفات الكلور التي تعمل على مستوى أدنى من الهيكل الهرمي لمراقبة الكمية والنوعية المتكاملة في شبكات توزيع مياه الشرب، مما يوفر حلاً عملياً لمراقبة نوعية المياه عبر الإنترنت في شبكات توزيع مياه الشرب. يتم تطوير نماذج المدخلات والمخرجات والحالة الفضائية لبقايا الكلور من النماذج الرياضية للديناميكيات المتبقية للكلور.
يتم توسيع خوارزمية تحليل المسار الحالية واستخدامها للحصول على هيكل المعلمات. يتم تطوير تقدير خطأ البارامتر المشترك والبنية النموذجية باستخدام نهج التقييد على أساس نموذج محدد النقطة.
يتم تحديد نصف قطر عدم اليقين للنظام من خلال تنبؤ قوي بالإخراج، يتم من خلاله فرض متطلبات دقة النموذج من التحكم التنبؤي النموذجي القوي صراحة على تقدير النموذج. وبالتالي، يتم تحقيق تصميم متكامل لتقدير وحدة التحكم والنموذج. يطبق التحكم التنبؤي النموذجي للتحكم المتبقي من الكلور استنادا إلى النموذج المحدد. ولتلبية القيود المفروضة على النواتج في ظل أوجه عدم اليقين في النظام، تُستخدم مناطق الأمان، المصممة من خلال تقييم إلكتروني لسيناريوهات عدم اليقين في النظام، لتقييد قيود النواتج.
يمكن الحصول على مناطق الأمان عن طريق حل مشكلة تحسين غير خطية مقيدة باستخدام خوارزمية استرخاء واكتساب مبسطة بشكل كبير. ويتم التحقق من المنهجية المقترحة بتطبيقها على نظم محاكاة لتوزيع مياه الشرب.
تتم مناقشة دراسة محاكاة لتقدير النموذج وعملية التحكم التنبؤي النموذجي.



**كلمات مفتاحية**: أنظمة توزيع مياه الشرب، الكشف عن أخطاء أجهزة الاستشعار، تحليل المكون الأساسي غير الخطي ، الشبكة العصبية المختنق ، نموذج تاكاجي سوجينو الضبابي ، التحكم التنبئي للنموذج الضبابي.

# « Diagnostic et Contrôle des Systèmes de Distribution de l'eau Potable »

**Résumé** :

Le succès de toute stratégie de diagnostic dépend essentiellement des capteurs qui mesurent les variables de processus. Cette thèse présente une méthode de détection et de diagnostic des anomalies des capteurs basée sur un réseau neuronal goulot d'étranglement (BNN). L'approche BNN est utilisée comme outil de contrôle statistique des processus pour les réseaux de distribution d'eau potable (DWDS) afin de détecter et d'isoler les défauts des capteurs. Cette méthode est validée en simulation sur un système non linéaire: un réservoir quadruple et un système de distribution d'eau potable réel. L'objectif de cette thèse est de contribuer à la supervision de DWDS en utilisant la méthode de diagnostic du réseau neuronal. Considérant la forte demande d'approvisionnement en eau en raison de la croissance démographique, l'approbation de nouvelles technologies pour assurer la qualité de l'eau à moindre coût est essentielle. Cette thèse présente un DWDS basé sur une technique de modélisation floue non linéaire. L'approche utilise un modèle flou de Takagi-Sugeno (T-S) multiinput multi-output (MIMO), qui est pertinent pour construire une grande classe de processus non linéaires. Le cadre proposé est validé sur un véritable réseau de distribution d'eau potable, le modèle T-S flou MIMO était mise en œuvre, dans le contexte d'un contrôle prédictif non linéaire pour réguler la qualité de l'eau (la concentration du chlore dans l'eau potable). L'objectif est de maintenir les sortis du système entre les limites supérieures et inférieures par rapport aux exigences des règlements sanitaires. Les modèles entrées-sorties et espaces d'états des résidus du chlore sont élaborés à partir de modèles mathématiques de la dynamique résiduelle du chlore. L'estimation des erreurs de paramètres et de structures de modèles est élaborée à l'aide d'une approche limitative fondée sur un modèle paramétrique ponctuel. Le rayon d'incertitude du système est défini par une prévision robuste des extrants, grâce à laquelle les exigences de précision du modèle à partir d'un contrôle prédictif du modèle robuste (MPC) sont explicitement imposées à l'estimation du modèle. Par conséquent, une conception intégrée du contrôleur et de l'estimation du modèle est réalisée. MPC est appliqué pour le contrôle résiduel du chlore en fonction du modèle à limites fixes. Pour satisfaire aux contraintes de production dans le cadre des incertitudes du système, des zones de sûreté sont utilisées, conçues à partir d'une évaluation en ligne des scénarios d'incertitude du système, afin de limiter les contraintes de production. Les zones de sécurité peuvent être obtenues en résolvant un problème d'optimisation à contrainte non linéaire à l'aide d'un algorithme de relaxation-gain considérablement simplifié. L'étude de l'estimation du modèle de MPC est discutée.

**Mots clés** : Réseaux de distribution de l'eau potable, détection des défauts-capteurs, ACP Nonlineaire, BNN, Modèles Flou Takagi-Sugeno, Commande prédictive Floue .

# « Diagnostic et Contrôle des Systèmes de Distribution de l'eau Potable »

**Abstract** :

The sensors measuring process variables are important to the success of any diagnosis technique. This thesis describes a method for detecting and diagnosing sensor problems using a Bottleneck Neural Network (BNN). To detect and isolate sensor problems in drinking water distribution systems (DWDS), the BNN technique is employed as a statistical process control tool. This method has been validated in simulation on a nonlinear system: a quadruple tank and a real-world drinking water distribution system. The goal of this study is to make a contribution to the supervision of a DWDS utilizing the Neural Network diagnosis approach. With the increased demand for water owing to population expansion, the approval of new technologies to ensure water quality at a lesser cost is critical.

The DWDS presented in this thesis is based on a nonlinear fuzzy modeling technique. The method employs a multi-input multi-output (MIMO) Takagi-Sugeno (T-S) fuzzy model, which is useful for building a wide range of nonlinear processes. The suggested framework was validated on a real drinking water distribution system by implementing the MIMO fuzzy T-S model in the context of nonlinear predictive control to regulate water quality (the chlorine concentration in drinking water). The goal is to keep the system outputs within the maximum and lower limitations set by health laws. Online feedback control of chlorine residuals at the lowest level of a hierarchical structure of integrated quantity and quality control in DWDS, providing a viable solution for online water quality control in DWDS. From mathematical models of chlorine residual dynamics, input-output and state-space models of chlorine residuals are constructed. To acquire the parameter structure, the existing path analysis algorithm is enhanced and used. A bounding approach based on a point-parametric model is used to estimate joint parameter and model structural errors.

The system's uncertainty radius is defined by robust output prediction, which explicitly imposes model accuracy criteria from robust model predictive control (MPC) on model estimation.

As a result, an integrated controller and model estimation design is realized. Based on the set-bounded model, MPC is used for chlorine residual control. To fulfill output limits under system uncertainties, safety zones are used, which are developed based on an online evaluation of the system's uncertainty scenarios. Solving a nonlinear restricted optimization problem with a much simplified relaxation-gain approach yields the safety zones. The proposed methodology is validated by using a simulated DWDS. It is discussed how to conduct a simulation study of model estimate and MPC operation.

**Key words** : Drinking Water Distribution Systems (DWDS), sensor fault detection, Non-linear PCA, BNN, Takagi-Sugeno Fuzzy Model, Fuzzy Model Predictive Control

*For the spirit of my father, my hero.*

# Acknowledgements

All the praises and thanks be to God, who has guided me to this, for letting me through all the difficulties. I will keep on trusting Allah for my future.

My sincerest gratitude is given to my supervisor Prof. Salah CHENIKHER who gave me the opportunity to do this project, I would like to acknowledge and give my warmest thanks to my Co-Supervisor Prof. Messaoud RAMDANI who made this work possible. His professional guidance, patience and advices carried me through all my PhD research.

I am thankful to Prof. Nasr Eddine DEBBACHE for being the president of the jury committee of my thesis defense. I would like to thank Prof. Khaled KHELIL and Prof.Sofiane GHERBI for agreeing to attend to my thesis defense as jury members. I would also like to thank and all members of the jury for their valuable comments towards the improvement of this thesis.

I am grateful to my parents, my brothers NASR EDDINE and HATEM also all my family members and my friends for their unconditional love and support during this years. This doctoral thesis is dedicated to them.

# Table of contents

**List of figures** **ix**

**List of tables** **xi**

List of abreviations . . . 1

**1 Introduction** **3**

1.1 Objective of the project . . . 6

1.2 Thesis sturcture . . . 8

**2 Analytic Model of Drinking Water Distribution Systems** **11**

2.1 Introduction . . . 11

2.2 Physical Laws . . . 12

2.3 Hydraulic Laws in Drinking Water Distribution Networks . . . 13

2.3.1 Pumps . . . 14

2.3.2 Pipes . . . 15

2.3.3 Valves . . . 15

2.3.4 Reservoirs . . . 17

2.3.5 Nodes . . . 18

2.4 Quality Model . . . 18

2.5 Simulation Study . . . 19

2.5.1 Water Network Simulator: EPANET . . . 19

2.5.2 Simulation Implementation in Matlab . . . 21

2.6 Summary . . . 23

**3 Empirical Modeling** **25**

3.1 Introduction . . . 25

3.2 Empirical Modeling Principles . . . 27

3.3 Neural Networks Approach . . . 27

3.4 Fuzzy Clustering . . . 29

3.4.1 Hard Clustering . . . 32

3.4.2 Fuzzy Partition . . . 35

3.4.3 Possibilistic Clustering . . . 37

3.4.4 Fuzzy C-means Clustering . . . 37

3.5 MIMO FUZZY MODEL . . . 46

3.5.1 Takagi-Sugeno Fuzzy Modeling for multivariable nonlinear system . . . 46

3.6 T-S FUZZY MODEL IDENTIFICATION . . . . . 47
3.7 State Space Model . . . . . 49
3.8 Summary . . . . . 51

**4 Model based Predictive Control** **53**
4.1 Introduction . . . . . 53
4.2 Formulation of Model Predictive Controller . . . . . 54
4.3 Takagi-Sugeno Fuzzy Model Predictive Control . . . . . 59
4.4 Summary . . . . . 61

**5 Data-driven based DWDS Monitoring** **63**
5.1 Introduction . . . . . 63
5.2 Linear Principal Component Analysis . . . . . 64
5.3 Nonlinear Principal Component Analysis . . . . . 65
5.4 The ConceptT Of Bottleneck Neural Network and Nonlinear Variable Reconstruction . . . . . 66
5.4.1 Fault detection . . . . . 68
5.4.2 Fault identification . . . . . 70
5.5 Summary . . . . . 73

**6 Experimental Results** **75**
6.1 Sensor Fault Detection and Isolation :Case Study . . . . . 75
6.2 Sensor Validation using BNN and Variable Reconstruction :Case Study . . . . . 81
6.3 Fuzzy Predictive Control of Chlorine Concentration in DWDS :Case Study . . . . . 85
6.3.1 Summary . . . . . 89

**Conclusion** **91**

**References** **93**

# List of figures


2.1 A Benchmark Drinking Water Distribution Network . . . 11
2.2 Physical Components in a Water Distribution System . . . 13
2.3 Water Pump . . . 14
2.4 Example of Pipe Network . . . 15
2.5 Model of Flow Control Valve . . . 17
2.6 Model of a Reservoir . . . 17
2.7 Chlorine Paths in a Simple Water System . . . 19
2.8 Procedure of Chlorine Model Parameter Estimation . . . 21
2.9 Experiment Implementation for Model Parameter Estimation . . . 22
2.10 Simulation Environment Implementation . . . 23

3.1 Clusters of different shapes and dimensions . . . 31
3.2 A data set in $\Re^2$ . . . 34
3.3 Different Distance Norms used in Fuzzy Clustering . . . 44
3.4 The fuzzy c-means clusters, .. are the data points, + are the cluster means . . . 45

4.1 Operation of the MPC . . . 55
4.2 Zone Reference MPC . . . 58

5.1 Bottleneck Neural Network . . . 66
5.2 The architecture of the NLPCA based on RBF networks . . . 67
5.3 Bottleneck Neural Network for Fault Detection . . . 69
5.4 Variable reconstruction in the linear case . . . 71
5.5 Nonlinear Reconstruction . . . 71

6.1 A schematic picture of the quadruple tank process . . . 76
6.2 SPE normal case . . . 78
6.3 (a)The statistical SPE of the global PCA for the abnormal state. (b)The contribution of the variables . . . 78
6.4 The water distribution network under study . . . 79
6.5 SPE normal case . . . 80
6.6 (a)The statistical SPE of the global PCA for the abnormal state. (b)The contribution of the variables . . . 80
6.7 Schematic representation of the aerated bioreactor for wastewater treatment . . . 82

6.8 The reconstruction of the variables in the fault free case. The variables $x_j$ and their reconstructed values $\hat{x}_j$, $j = 1,\ldots,5$. . . . . 82
6.9 SPE normal case . . . . . 83
6.10 SPE for faulty case . . . . . 84
6.11 SPE after reconstruction of variables . . . . . 84
6.12 A Benchmark Drinking Water Distribution Network . . . . . 86
6.13 Time series of chlorine injection and concentration . . . . . 87
6.14 Performance of the T-S fuzzy model of chlorine concentration in outputs nodes, $'-'$ data and$'-$ $-'$model estimation. . . . . 88
6.15 Setpoint tracking performance of MPC based on T-S fuzzy model . . . . . 89

# List of tables

6.1 parameter values of the quadruple-tank . . . . . 77
6.2 Operating points parameters . . . . . 77
6.3 T-S fuzzy model performances . . . . . 87

# List of abreviations

BNN : Bottleneck Neural Network
**DTW** : Dynamic Time Warping
**DWDS** : Drinking Water Distribution System
**EM** : Empirical Modeling
**FDD** : Fault Detection and Diagnosis
**HLA** : Hybrid Learning Algorithm
**HMM** : Hidden Markov Models
**MIMO** : Multi Input Multi Output
**MPC** : Model Predictive Control
**MSPC** : Multivariate Statistical Process Control
**NLPCA** : Nonlinear Principal Component Analysis
**PCA** : Principal Component Analysis
**PLS** : Partial Least Squares
**RBF** : Radial Basis Function
**RHC** : Receding Horizon Control
**RMSE** : Root Mean Square Error
**SPE** : Squared Prediction Error
**SVI** : Sensor Validity Index
**T-S** : Takagi-Sugeno
**WLS** : Weighted Least squares
**WRLS** : Weighted Recursive Least Square

# Chapter 1

# Introduction

Nowadays, it is possible to gather enormous amounts of measured data because to the substantial advancements in process sensing technology. However, the useful data about the process's state is rarely applied in the best possible way. The difficulty of digesting the information is a common cause of this improper management of the process measurements. In order to completely utilize and accommodate automated in-process sensing devices and collect all diagnostic data linked to quality, it is important to establish an effective technique.

The concept of information redundancy is utilized by modern model-based condition monitoring. In reality, anomalous states can be identified by comparing observed behavior reported by sensors to expected behavior predicted by mathematical models to determine whether there is any consistency.

Models can be generally explicit, developed from fundamental ideas or system identification[19], or implicit, derived via principle component transformation[33]. Some structured residuals for fault isolation that react to subsets of faults may be produced directly or through algebraic transformation[20].

Structured partial principal component models are used for the direct computation in the principle component framework[11]. In the PCA framework, further fault isolation methods include sensor validity indices, statistical measurements, and contribution charts[33]. Multivariate statistical process control, also known as MSPC, is a statistical methodology that has been utilized in process monitoring during the past ten years to regulate and enhance industrial processes across a variety of industries[39]. By defining a smaller set of artificial variables, MSPC tries to decrease the redundancy that is frequently observed in the recorded variables[16].

PCA is a frequently employed algorithm within the realm of Multivariate Statistical Process Control (MSPC). Subsequently, the process of identifying and addressing faults and anomalies (FDD) takes place in a reduced-dimensional space. This involves vigilant monitoring of charts depicting principal component scores as well as charts illustrating the sum of prediction errors (SPE) as detailed in[33].

Following the principles of PCA theory, the loading plot becomes instrumental in revealing the interplay among the initial variables. This provides a means to detect anomalies as discrepancies often disrupt relationships between these variables. Regrettably, the conventional application of PCA tends to yield an abundance of erroneous alerts or, conversely, overlooks the identification of process malfunctions. This is especially pronounced when a process encompasses multiple operational states, significantly compromising the reliability

of the monitoring system. Substantial research efforts have been dedicated to mitigating these constraints, yielding various approaches aimed at circumventing these challenges.

Hastie[22] introduced the concept of principal curves to extend PCA non-linearly, while Kramer[27] employed auto-associative neural networks, and Qin and McAvoy[38] integrated neural networks into the partial least squares (PLS) framework. Webb[48] and Wilson[50] separately proposed nonlinear extensions of PCA utilizing radially symmetric kernel functions and radial basis function (RBF) networks, respectively.

Recent advancements in sequential data analysis have led to the development of two techniques, namely Dynamic Temporal Warping (DTW) and Hidden Markov Models (HMM), designed for effective Fault Detection and Diagnosis (FDD). In addition to these, suggestions have emerged advocating the use of empirical models like fuzzy logic and neural networks. While the role of the fuzzy logic method is pivotal, it also serves as a foundational framework for integrating other approaches. The evolution of technology and the adoption of advanced data collection techniques have necessitated the simultaneous monitoring of a greater number of variables.

Multivariate Statistical Process Control (SPC) charts provide a swift means of identifying process disturbances. However, it's common that process operators become aware of assignable causes leading to multivariate process deviations only when such disruptions are highlighted by these charts. This insight is drawn from Shao's work in[42].

The rising global demand for water, attributed to population growth, industrial progress, and enhanced economic status, is met with a reduction in both the count and capacity of available water sources. Moreover, endeavors involving water handling and conveyance entail significant energy consumption. Amid this scenario, the effective exploitation of water supply networks emerges as the most viable approach. Consequently, the efficient management of water distribution transfers becomes imperative[35].

This study aims to make a valuable contribution to the oversight of water distribution network systems. It does so by employing the principal component analysis method, offering a systematic approach to enhance the supervision of these complex systems.

Although detecting novel occurrences is a key ability of any signal classification method, it is an incredibly difficult assignment. As a result, multiple novelty detection models have been demonstrated to work effectively on various data sets. It is clear that there is no one best model for novelty detection, and success is dependent not just on the approach utilized, but also on the statistical features of the data[32].

In bioprocessing, multivariate statistical process control (MSPC) has been applied. When typical MSPC methodologies are applied to industrial processes for performance monitoring, however, they are not always as effective as intended due to the unfitness of the process statistical model. In many cases, the observed process signals contain a significant amount of measuring or operating noise, making it difficult to extract the true signals from the noise-corrupted process signals using traditional latent projection methodologies such as principal component analysis (PCA) or partial least squares (PLS)[30].

In this work, we offer another type of MSPC approach that leverages nonlinear variable reconstruction process signals collected from observed process data to increase process monitoring performance. The concept is

based on employing Nonlinear Principal Component Analysis (NLPCA) based on Bottleneck Neural Network (BNN) to collect process blind signals and Squared Prediction Error to eliminate process faults.

The water distribution systems have been modeled extensively in order to improve their performance. There have been numerous mathematical models developed, the majority of which are based on mass, energy conservation laws, and data-driven models.

Analyzing both the quantity and quality of water is essential for the effective operation of Drinking Water Distribution Systems (DWDS). Consequently, the development of a suitable model plays a pivotal role in comprehensively evaluating water distribution networks[10].

Duzinkiewicz.K [17]proposed an optimization-centered set-membership strategy for establishing bounded estimates of water quantity and quality within Drinking Water Distribution Systems (DWDS). However, this approach's time-intensive nature prompts consideration of an alternative solution. Consequently, addressing this challenge necessitates the exploration of a non-linear and non-convex optimization problem.

In pursuit of this objective, an interval observer is under development to serve the purpose of guaranteeing water quality[28].

Additionally, a considerable body of literature encompasses models related to residual chlorine degradation across water distribution networks, along with the placement of booster chlorination stations[24]. To strategically position suitable booster chlorination stations within Drinking Water Distribution Systems (DWDS) while maintaining the chlorine residual within specified limits, an innovative approach is introduced. This approach combines a hybrid Particle Swarm Optimization (PSO) technique with genetic algorithms to achieve an optimal solution.

Continuous validation of the resilience and security of water supply systems necessitates the efficient operation of sensors and actuators, ensuring their optimal functionality[8]

For the purpose of quality control within Drinking Water Distribution Systems (DWDS), the adoption of Model Predictive Control (MPC) is favored. This choice is attributed to the non-linear dynamics inherent in the quality model and the multifaceted challenges of constrained control in a multivariable context[51].

Rafal Langowski& al.[29] proposed an interval estimation technique to enhance quality monitoring within Drinking Water Distribution Systems (DWDS). This method incorporates an interval observer, enabling the derivation of robust interval boundaries for the anticipated water quality state variables .

## 1.1 Objective of the project

water is indeed a crucial resource for both domestic and industrial purposes. The availability of clean and safe drinking water is vital for sustaining human life and supporting various industries.

Drinking water is typically sourced from ground sources such as rivers, lakes, and underground aquifers (wells and springs). After extraction, it goes through a series of treatments to ensure it meets the necessary quality standards. This treatment process usually involves steps like filtration, disinfection (often with chlorine or other chemicals), and sometimes additional treatments like fluoridation for dental health.

Once the water is treated and safe for consumption, it needs to be efficiently transported from the water treatment plant to the end consumers. This is where the Drinking Water Distribution Systems (DWDS) come into play. DWDS are complex and large-scale networks that consist of various components like pumps, valves, storage tanks, and pipes made of different materials and varying diameters. DWDS are large-scale networks made up of diverse components such as pumps, valves, storage tanks, and pipelines constructed of varied materials and diameters. The design of a DWDS is critical in ensuring that it can deliver sufficient water quantity while maintaining acceptable water quality to meet consumer needs. Population density, water use patterns and possible future growth should all be considered.

The DWDS must also be maintained and monitored on a regular basis to prevent leaks, ensure water quality, and solve any issues that may emerge.

Overall, a well-designed and well-maintained DWDS is vital in providing communities and enterprises with safe and dependable drinking water.

Maintaining appropriate chlorine residual concentration limits is an important part of quality control for Drinking Water Distribution Systems (DWDS). The residual chlorine content is significant because it helps to disinfect the water and prevent the growth of dangerous microorganisms as it travels through the distribution network.

In this context, the control challenge is to regulate the chlorine concentration throughout the water network to keep it within certain lower and higher bounds. However, many obstacles challenge this control problem:

-Large Time Delays: Water may take a long time to travel from the treatment plant to the consumers through the distribution system. These temporal delays can have an effect on the chlorine concentration’s reaction to control activities.

-Constraints: To maintain compliance with safety and regulatory standards, the control system must consider multiple constraints such as the maximum and minimum allowed chlorine concentrations.

-Uncertainties: DWDS are subject to uncertainties, which may arise from a variety of causes, including variations in water demand, fluctuating water quality from different sources, and uncertainties in the distribution network's hydraulic behavior.

To create an effective controller, it is necessary to create a model that effectively represents the dynamics of chlorine concentration in the DWDS. This model should take into account elements such as the rate of reaction, transport phenomena, and mixing processes that influence changes in chlorine concentration over time.

Furthermore, because uncertainties are inherent in the system, appropriate tools to predict and manage these uncertainties are required. This entails making informed decisions and adapting to changing situations using past knowledge of the system, historical data, and real-time online observations.

The controller's primary goal is to ensure that the chlorine residual concentration remains within the acceptable boundaries in the presence of uncertainties. This may need the use of robust control strategies that can handle uncertainties, as well as advanced control approaches such as Model Predictive Control (MPC), which may optimize control actions over a predictive horizon while taking system restrictions into account.

Overall, an efficient control system for maintaining chlorine residual content in DWDS should be capable of dealing with time delays, handling uncertainties, and adhering to restrictions, all while providing safe and dependable water quality for users. This is critical for ensuring that clean and safe drinking water is available throughout the distribution network.

## 1.2 Thesis sturcture

The mathematical modeling of the chlorine residual concentration is discussed in Chapter 2. The hydraulic effects on the quality model are studied starting with the basic DWDS component model. It is possible to obtain mathematical models of chlorine transportation, mixing, and reaction kinetics.

The methods for obtaining input-output and state-space models are described. The DWDS input-output model structure with switching tank is investigated. In that chapter, a control problem is also studied. The path analysis approach, which is used to calculate the time delay in the chlorine transportation from the injection node to the monitoring node, is described next in order to identify the model parameter structure.

Chapter 3 delves into the T-S fuzzy modeling concept, with the system construction carried out through a two-part approach involving clustering for an initial coarse model, subsequently refined. The initial phase entails clustering the input space, followed by the application of a hybrid learning algorithm (HLA) in the subsequent phase. This HLA consists of a gradient descent-based optimization strategy for premise parameters and utilizes weighted least squares (WLS) for subsequent parameter estimation.

Central to this identification methodology are two defining features: a self-adaptation mechanism governing learning rates and the incorporation of a scaled metric norm to gauge distance. The outcomes of experimental evaluations underscore the high efficiency of the proposed solution.

The model predictive control (MPC) technique is described in Chapter 4. The DWDS mathematical model has been established. Controlling the chlorine concentration in drinking water networks is successfully accomplished using limited multivariable fuzzy predictive control. The control signals are computed to enhance the future operational performance of the underlying processes, focusing on achieving precise adherence to set-points.

The development of the Takagi-Sugeno fuzzy system follows a two-step process: first, the identification of its structure, and second, the adaptation of its parameters. The learning algorithm adapts the learning rates based on the evolution of the cost function. The simulation findings validate the proposed fuzzy control approach for regulating the chlorine concentration in the DWDS.

The fundamentals of multivariate statistical process monitoring are presented in Chapter 5, with a special emphasis on the usage of the Bottleneck Neural Network (BNN) in autoassociative mode to execute nonlinear principal component analysis (NLPCA). A fuzzy clustering-based projection strategy for NLPCA learning is introduced. To adapt this diag-

nostic method to nonlinear systems, a contribution is proposed to replace the linear PCA model with a Bottleneck Neural Network (BNN). The state reconstruction method is used to isolate sensor faults.

The simulation results are presented and analyzed in Chapter 6 of the thesis, The results and discussions help to validate the research and contribute to the field's understanding of water distribution system control and water quality management.

# Chapter 2

# Analytic Model of Drinking Water Distribution Systems

## 2.1 Introduction

A water distribution system is an integrated network of components that transports water from its source to many endpoints or customers. It is a necessary infrastructure that ensures the delivery of sufficient quantity and quality of water to meet user needs[21]. They are composed of active parts (such as valves and pumps) that can change the flow rate of water in the system and passive components (such as pipes, tanks, and reservoirs)[7]. Figure2.1

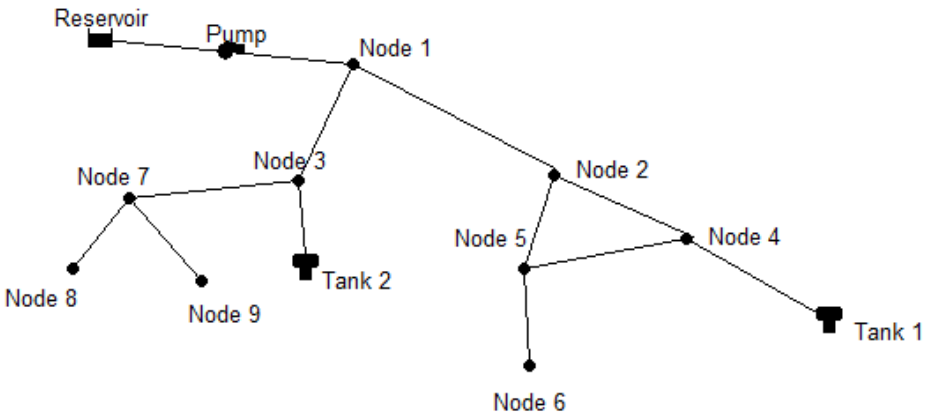


Fig. 2.1 A Benchmark Drinking Water Distribution Network

The mathematical model of chlorine concentration in the DWDS will be established in this chapter.

## 2.2 Physical Laws

- Conservation of Mass

The law of conservation of mass, often known as the principle of mass conservation, says that in a closed system, mass cannot be created or destroyed. This means that, independent of any physical or chemical processes occurring within the system, the total mass of a closed system remains constant over time.

When analyzing the input and outflow of mass at a junction node in a system, such as in fluid dynamics or transportation networks, the principle of mass conservation can be utilized.

The rate of mass storage inside a junction node (change in mass within the node over time) equals the difference between total mass input and total mass outflow from that node.

$$\frac{dS_i(t)}{dt} = \sum Q_i(t) - V_i(t) \tag{2.1}$$

where $Q_i$ is the total inflow into node $i$ $[m^3/s]$, $V_i$ the water volume at node $i$ $[m^3/s]$, and $dS_i/dt$ is the change in storage $[m^3/s]$.

- Conservation of Energy

Assume a pipe section with length $l$ $[m]$, cross sectional area $A$ $[m^2]$, and $\Delta h$ the head difference between the two ends of the pipe, $g$ the gravitational acceleration, and $h_loss$ the head loss through component $i$ along the path. The flow $Q_i(t)$ $[m3/s]$ evolution over the pipe is given by:

$$\frac{dQ_i(t)}{dt} = \frac{gA}{l}\left(\Delta h(t) - h_{loss}(t)\right) \tag{2.2}$$

Head losses represent the energy losses along the piping section, are designed as follows:

$$h_{loss}(t) = h_{loss.f}(t) + h_{loss.m}(t) \tag{2.3}$$

where $h_{loss.f}$ represents friction losses and $h_{loss.m}$ is the minor local losses[53]. To obtain a dynamic model of DWDS, parameters of network elements must be known or estimated, equations of mass and energy are used to develop a state space representation of the following form:

$$\dot{x}(t) = f\left(x(t), u(t), d(t)\right) \tag{2.4}$$

We consider that: $x$ is the state(the flow across pipes and the heads in nodes), $u$ is the control input(loss coefficient of valves and pressure injection of pumps), $d$ is the exogenous disturbance input, and $f$ is the nonlinear state transition function.

## 2.3 Hydraulic Laws in Drinking Water Distribution Networks

Water is the carrier of the chlorine in the distribution network. The behavior of chlorine transportation and mixing is directly influenced by specific factors such as water flows, velocities (speeds of water movement), and detention times (the duration water remains in a reservoir). These parameters play a crucial role in how chlorine is dispersed and mixed within the system.

The hydraulic properties of the system, which include the movement and distribution of water, have a substantial impact on the modeling of chlorine concentration. The way water flows and interacts with the environment directly affects how chlorine is distributed and diluted. Therefore, understanding the hydraulics is crucial for accurate modeling of chlorine concentration. The thesis presents a systems approach to modeling and operational control of water distribution networks. The hydraulic principles in DWDS provided here are useful in analyzing the sources of uncertainty in quality control. The description, however, is rather brief. Details are available in the chapter and in[28]

The water network is composed of various interconnected elements. These elements work together to facilitate the distribution and management of water within the network. They are divided into two groups based on flow rate: active elements (valves and pumps) that can adjust the flow rate of water in the system and passive components (pipes, tanks, and reservoirs).[13]. Figure 2.2

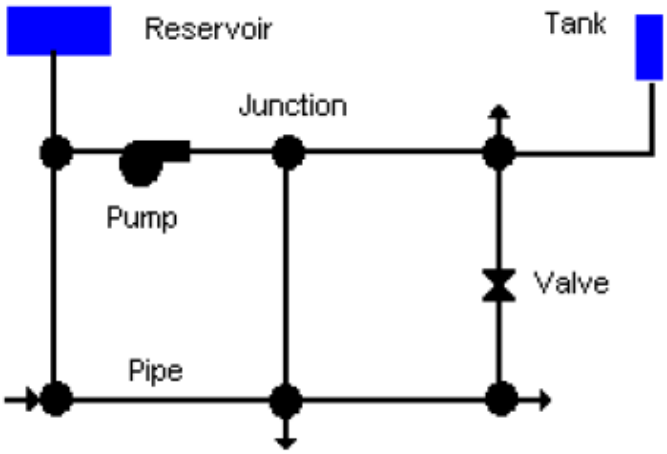


Fig. 2.2 Physical Components in a Water Distribution System

### 2.3.1 Pumps

pumps are crucial components of a Drinking Water Distribution System. They add energy to the system by increasing hydraulic head, facilitate water supply and pressure maintenance, and are primarily driven by electrical motors. Efficient management and control of pumps are essential for achieving the objectives of maintaining water pressure, optimizing energy consumption, and ensuring the effective operation of the DWDS. Figure 2.3.

The relationship between the head created by a set of variable-speed pumps running in parallel and the speed of the pumps and their output flow is described as nonlinear. In this context, the following variables are mentioned:

-Head $(h_p)$ $[m]$: This represents the hydraulic head generated by the pumps in meters. The hydraulic head is a measure of the potential energy of the fluid and is often associated with the pressure or elevation of the fluid within the system.

-Speed $(N)$ $[rpm]$: Speed refers to the rotational speed of the pumps, typically measured in revolutions per minute $(rpm)$. Variable-speed pumps have the ability to adjust their rotational speed, which affects their pumping characteristics.

-Output Flow $(Q_p)$ $[m^3/s]$: Output flow represents the volume of fluid (usually water) pumped by the system per unit of time. It is typically measured in cubic meters per second $(m^3/s)$.

$$h_p(t) = a_0 N(t)^2 + \frac{b_0}{n} N(t) Q_p(t) - \frac{c_0}{n^2} Q_p^2(t) \tag{2.5}$$

with $a_0$, $b_0$ and $c_0$ are the constants for a specific pump characteristics.

Fig. 2.3 Water Pump

### 2.3.2 Pipes

A pipe is an element of circular section, intended for transporting liquids (e.g.,water) from one junction to another as shown in figure 2.4. Take into account a pipe section with sectional area $A_p$ $[m^2]$, length $l_p$ $[m]$ and head difference between two ends of pipe $\Delta h_p$ $[m]$. The flow $Q_p(t)$ $[m^3/s]$ through a pipe is given by the following differential equation:

$$\frac{dQ_p(t)}{dt} = \frac{gA_p}{l_p}\left(\Delta h_p(t) - h_{loss}(t)\right) \tag{2.6}$$

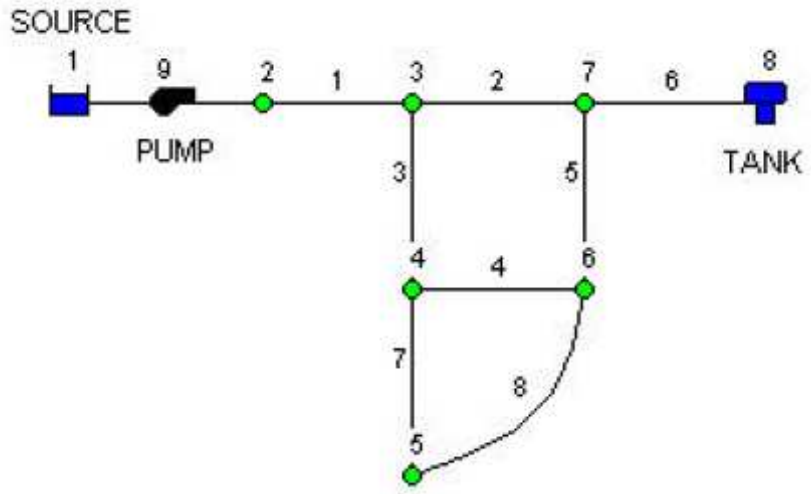


Fig. 2.4 Example of Pipe Network

### 2.3.3 Valves

Valves play a crucial role in a drinking water distribution system (DWDS), performing a wide range of functions that contribute to the overall operation, maintenance, and safety of the system. Here are some common types of valves found in DWDS and the functions they serve:

- Gate Valves: These valves control the flow of water by raising or lowering a gate (a wedge-shaped barrier). They are often used to completely shut off or allow full flow through a pipe.
- Ball Valves: Ball valves have a rotating ball with a hole that can be aligned with the pipe to allow flow or turned perpendicular to shut off the flow. They provide quick and reliable shutoff capabilities.

- Butterfly Valves: These valves consist of a circular disc mounted on a shaft. Rotating the disc controls the flow by varying the opening. They are often used for large-diameter pipes.
- Check Valves: Check valves allow water to flow in only one direction, preventing backflow and maintaining the direction of flow in the network.
- Pressure Reducing Valves (PRVs): PRVs maintain a consistent downstream pressure by reducing the incoming pressure from the supply. They ensure that pressure remains within safe limits throughout the system.
- Pressure Sustaining Valves (PSVs): PSVs maintain a minimum pressure in a specific zone, preventing pressure drops that could affect user service.
- Pressure Relief Valves: These valves release excess pressure from the system to prevent damage or over-pressurization.
- Air Release Valves: Air release valves expel air from the system to prevent air pockets that could disrupt flow and reduce system efficiency.
- Fire Hydrants: Although not strictly valves, fire hydrants are important components that allow water to be extracted for firefighting purposes.
- Isolation Valves: These valves are strategically placed to isolate sections of the network, making repairs and maintenance easier without affecting the entire system.
- Control Valves: Control valves regulate flow or pressure based on signals from sensors or controllers. They enable dynamic adjustments to optimize system performance.
- Throttle Valves: Throttle valves control flow by restricting the cross-sectional area of the pipe. They are used for fine-tuning flow rates.
- Mixing Valves: Mixing valves blend water from different sources or at different temperatures to achieve the desired water quality or temperature.

These are just a few examples of the many types of valves in a drinking water distribution system. Each type of valve serves a specific purpose in ensuring efficient, reliable, and safe water distribution to consumers. Proper selection, installation, and maintenance of valves are essential for the effective operation of the entire DWDS.

As a modelling example, the flow control valve is illustrated in Figure2.5 . The flow control valves is a kind of variable valves that can be modelled as a pipe with controlled conductivity, that is:

$$q_{ij} = V_{ij} G_{ij} \left(h_i - h_j\right) \left|h_i - h_j\right|^{-0.46} \tag{2.7}$$

where $V_{ij}$ denotes the controlled value and the valve is closed if $V_{ij} = 0$ and fully opened if $V_{ij} = 1$ .

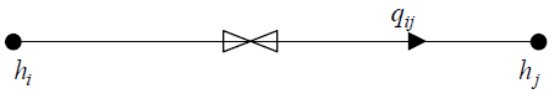


Fig. 2.5 Model of Flow Control Valve

### 2.3.4 Reservoirs

A reservoir is a natural or manufactured place where water is gathered and stored for use. The reservoir head without external pressure at the point of the discharging is represented in the next simplified continuity equation :

$$\rho \frac{dh(t)}{dt} = \frac{1}{c} \left[\rho_i Q_i(t) - \rho_0 Q_0(t)\right] \tag{2.8}$$

where $\rho$, $\rho_i$, $\rho_0$ are the water reservoir density, the inflow and outflow, respectively, and they are supposed equal. $Q_i(t)$ $[m^3/s]$ is the input flow rate and $Q_0(t)$ $[m^3/s]$ is the output water flow rate. $h(t)$ $[m]$ is the head and $c$ $[m^2]$ denotes the capacity of the reservoir. Figure2.6

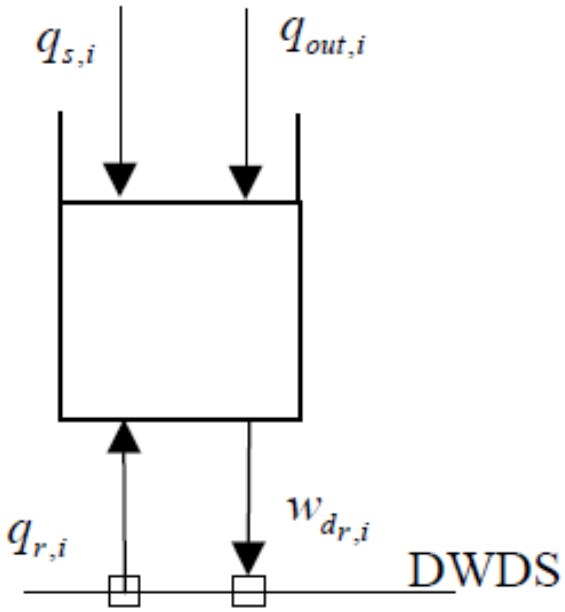


Fig. 2.6 Model of a Reservoir

### 2.3.5 Nodes

A node is an intersection in the water network that joins links together and where water enters or on the other side leaves the system. The law of conservation of mass is considered at nodes, it affirms that the rate of storage in a junctions node and tanks is equal to the difference between inflow and outflow.

$$\frac{dS_i(t)}{dt} = \sum Q_i(t) - V_i(t) \tag{2.9}$$

where $dS_i/dt$ is the change in storage $[m^3/s]$, $Q_i$ $[m^3/s]$ represents the total inflow in node $i$ and $V_i$ $[m^3/s]$ is the water volume used in node $i$[18].

## 2.4 Quality Model

Several chemical disinfectants, including as chlorine, which is affordable and simple to apply to water, can be used in the DWDS to eradicate the germs that cause diseases and illnesses. However, it should be kept within set limitations. When a chemical is incorporated into the water at a specific location in the DWDS, the flow of water carries it along with it. A quality model simulates how the chemical disperses as water travels through pipes, tanks, and reservoirs, indicating how the chemical concentration changes as it goes through the system[18]. The operational control of chlorine concentration in DWDS is based on a forecast of water demand for a minimum of a $24-hour$ period. An higher level controller creates the pump and valve operation schedules. The seasonal, climatic, and cultural patterns that influence water consumption might shift rapidly due to urgent situations like fighting fires.

In this chapter, the mathematical model of chlorine concentration in the DWDS is to be established.

Transportation, mixing, and reaction are the three factors that govern the chlorine residual concentration in the water network. Typically, chlorine is injected at some nodes in the distribution network and added to the water treatment process as shown in figure2.7, the source node of chlorine injection is node $A$ and monitored node is $G$, $y(t)$ and $u(t)$ denote the chlorine concentration at the monitored node and injection node, respectively, which are the model output and input.

Only chlorine residual concentration in the distribution network is considered in this thesis.

The following equation describes chlorine transfer and decay in a linear water flow:

$$\frac{\partial C_i(t,d)}{\partial t} = -v\frac{\partial C_i(t,d)}{\partial d} + R_i C_i(t,d) \tag{2.10}$$

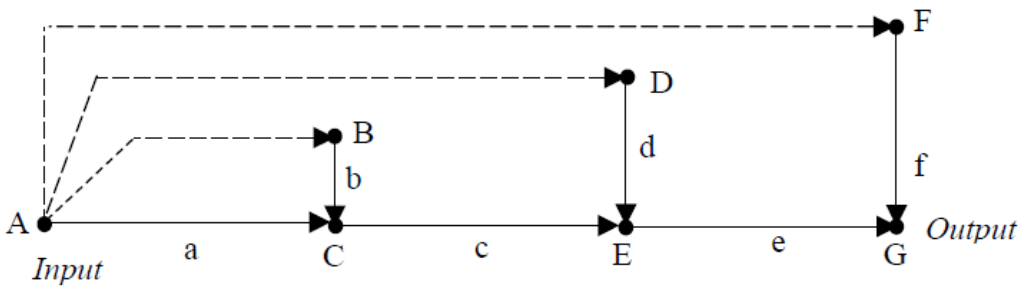


Fig. 2.7 Chlorine Paths in a Simple Water System

where $C_i$ denotes the chlorine concentration in pipe $i$, $t$ is the time instant during the hydraulic period, $v$ represents the velocity flow in pipe $i$, $d$ means the distance from the initial node, and $R_i$ indicates the reaction rate coefficient in pipe $i$.

The Lagrangian approach is a common technique used in software for modeling Drinking Water Distribution Systems (DWDS) to simulate the transport and concentration of chemicals, such as chlorine, in the water. This approach is widely employed because it provides a detailed and accurate representation of how chemicals move and disperse within the network, using a real measurements:

$$y(k) = f(F; d_c, u(k)) \tag{2.11}$$

where $y(k)$ is a vector of the chlorine concentration data at each node. The function $f$ represents the hydraulic and quality solution algorithm. This function takes as arguments the consumer demands $d_c$, graph $F$, which represents the nodes and pipes of the network.

## 2.5 Simulation Study

### 2.5.1 Water Network Simulator: EPANET

EPANET, a widely used computerized simulation model, was developed by the United States Environmental Protection Agency $(EPA)$. It is a software tool that was released in 2000 by the National Risk Management Research Laboratory. The software is open-source and free, and it is widely used in DWDS simulation and analysis of hydraulic and water quality behavior in pressurized pipe networks[40].

The fundamental characteristics and capabilities of EPANET are as follows:

-Hydraulic analysis: EPANET can simulate the hydraulic behavior of water flow within the distribution system, considering factors such as pipe characteristics, pumps, valves, stor-

age tanks, and demand patterns. It helps in assessing pressure levels, flow rates, and water age throughout the network.

-System optimization: EPANET can be utilized to optimize the operation and design of the water distribution system. It aids in the appropriate placement of pumps and storage tanks, pipe sizing, pressure control, and other system configurations in order to meet regulatory standards and increase operating efficiency.

-Reporting and visualization : EPANET provides graphical tool for visualization for displaying network layout, hydraulic results and water quality information. It also enables the generation of detailed reports that summarize the simulation results.

-Scenario analysis: EPANET enables utilities to assess the impact of system modifications or serious incidents on hydraulic performance and water quality by analyzing several scenarios. This helps in decision-making, emergency response planning, and assessing the efficacy of system modifications or interventions.

Water utilities can use EPANET to acquire useful insights into the performance of their distribution systems, optimize operations, guarantee regulatory compliance, and improve overall water quality management through the integration of EPANET. It is an effective technique for analyzing the transport and changes of drinking water inside a distribution system[13].

EPANET is a powerful software tool that enables a comprehensive understanding of drinking water distribution systems. Its adaptability allows it to be used for a wide range of applications, from hydraulic modeling and water quality analysis to system optimization and risk assessment. Water utility professionals, researchers, and engineers can utilize EPANET to make informed decisions, enhance system performance, and ensure the delivery of safe and high-quality drinking water to consumers[41].

In this thesis, EPANET is used to simulate the hydraulic and chlorine residual dynamics of the DWDS. The controls of pumps and valves in the DWDS are provided by an integrated quantity and quality controller at the upper level of a hierarchical structure as described in Chapter 2. Therefore, these operational controls are considered to be known. A DWDS network file is created based on the operational information. When running EPANET through the MS Windows graphic interface, the resulting hydraulics are saved in a flat file and can subsequently be read easily in Matlab for chlorine residual modeling.

The EPANET source code is integrated in a $C-file$ discrete $S-function$ for Matlab to simulate the chlorine dynamics. Iteratively calling the chlorine residual dynamic solver is a new feature of EPANET. Thus, the hydraulics over the $24-hour$ simulation horizon are solved first in the initialization component of the S-function, and then the chlorine dynamics can be solved iteratively at each time step of the discrete $S-function$. At each time interval

of the S-function, chlorine injections are updated by getting the S-function input values from the RMPC block. The S-function outputs show the chlorine concentrations at the monitored nodes[41].

### 2.5.2 Simulation Implementation in Matlab

Model parameters derived from historical data are insufficient for describing current and future system dynamics. The fundamental reason is that water demand does not repeat itself every day. As a result, operational controls and hydraulics are not repeated[41].
Measurements (historical data) are used to develop implicit models. As a result, numerical simulators for hydraulic and quality information (implicit models) are required to forecast real plant outputs for use in model parameter estimation for input-output or state-space models (explicit models).
IO or SS models are explicit models that are beneficial for control design but involve the use of additional parameters. The model parameters can be acquired using the numerical results of the implicit model. The overall procedure is illustrated in Figure2.8 [13]

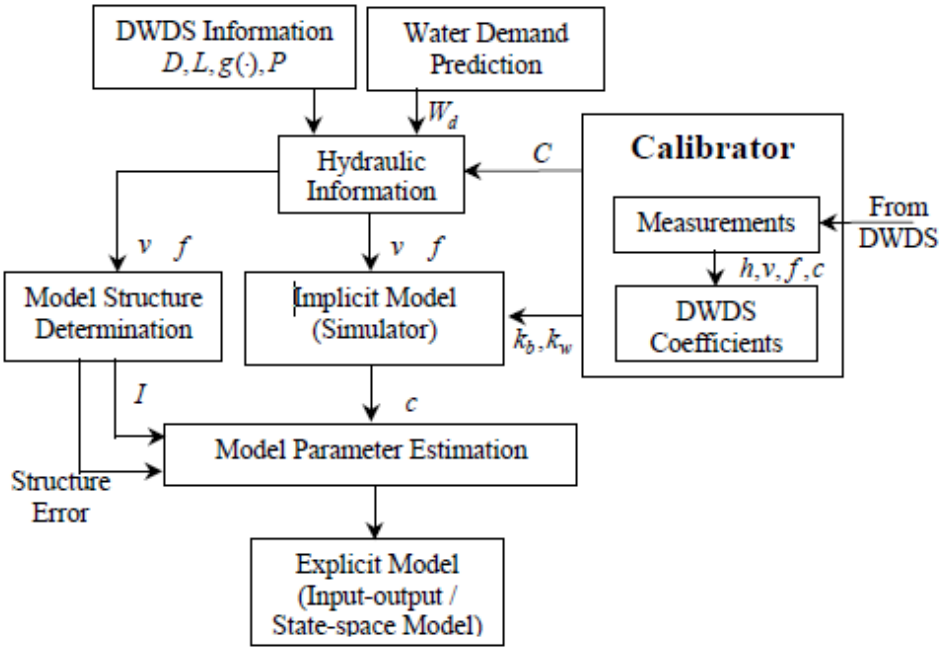


Fig. 2.8 Procedure of Chlorine Model Parameter Estimation

The figure shows that the hydraulic models must be established in advance. Where a number of coefficients are required, a quantity and quality simulator for the DWDS can be constructed. Some coefficients can be measured directly and remain constant during DWDS operation. Some are calculated from other known values. Others can only be approximated online or offline using measurement data and must be calibrated on a regular basis.

There are mainly two steps in the case study. First is to obtain the model of the chlorine residual of the DWDS, which could be IO type or SS type. Quantity and quality data are needed in order to perform path analysis algorithm and model parameter estimation. Path analysis requires hydraulic data of water flow and velocity in the network, which can be obtained by running EPANET in Windows and saving the simulation results into files that are read by path analysis algorithm implemented in Matlab.

To generate the data needed for model parameter estimation. The experiment is implemented in Matlab by embedding the EPANET source codes into a S-function, which is illustrated in Figure2.9 [13].

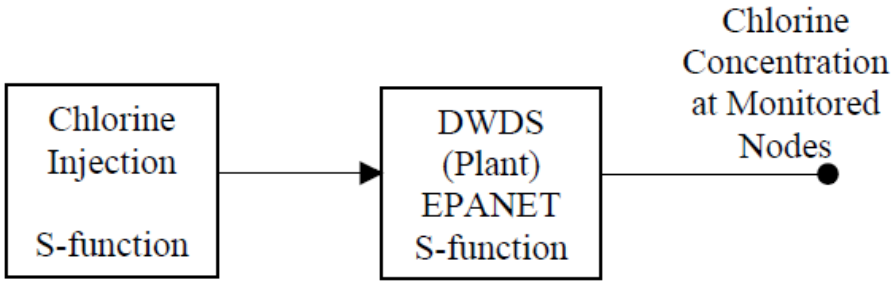


Fig. 2.9 Experiment Implementation for Model Parameter Estimation

Next, Model Predictive Controller is designed based on the state-space model as show in chapter 4. The Matlab simulation implementation of the case study is illustrated in Figure 2.10

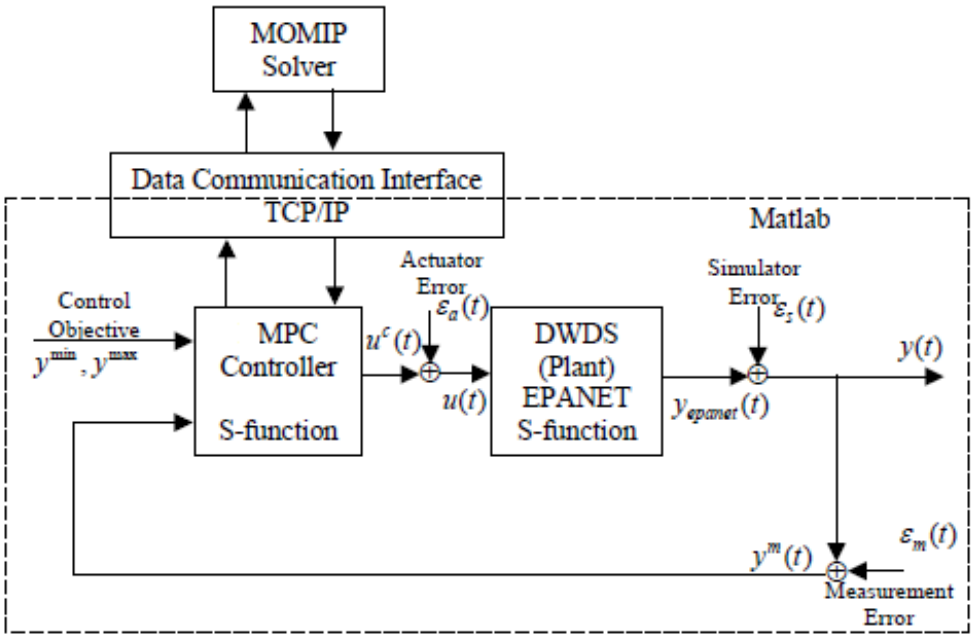


Fig. 2.10 Simulation Environment Implementation

## 2.6 Summary

The input-output and state-space models described in this Chapter are appropriate for control applications. Data obtained by the implicit model (plant simulator) can be used to estimate their parameters.Considering the network flows, the model structure can be deduced in either IO or SS form. Though the quantity control inputs are known when the quality controller is designed, the corresponding network flows are not. They can, however, be predicted using a network simulator with predicted disturbance inputs, such as water demand. The chlorine residual model structure design procedure can be explained as follows:

1. Implement a DWDS hydraulic model (quantity simulator).
2. Path analysis is used for determining the structure of an IO or SS model.

The parameters of the model must be estimated in order to complete the design. This requires chlorine measurements above the control horizon. They are, however, unavailable at the time the parameter estimation is being performed. We can try to use previous measurements, which will work if water demand across the future control horizon does not differ significantly from the demand pattern in the past, and if the quantity controls in the past are similar to the ones that will be used now. In practice, such a situation is very rare. It is proposed once more to use the network simulator with expected demand to generate future chlorine residuals and consider such chlorine residuals as pseudo measurements. The model parameters will be estimated using the pseudo measurements.

The water network coefficients that the simulator uses are stationary. As a result, they can

be calculated and calibrated using historical measurement data. Therefore, the following procedures are taken to finish estimating parameters in models for control purposes:
3. In DWDS (quality simulator), an implicit model of chlorine residuals is constructed.
4. Finally, the model parameters are established.
Steps 1 and 3 include implementing a numerical model of chlorine residuals. In order to implement such a model, DWDS coefficients such as pipe length, diameter, roughness, tank characteristics, chlorine reaction constant, and water demand are required. In the thesis, these processes are referred to as implicit modeling since they are implicit in describing the input-output relationship required by control design and operation. The implicit modeling is well developed, and there is a substantial literature on the subject. The last phase of the preceding approach is the focus of this chapter's discussion of model parameter estimation. In order to get the chlorine residual model, a number of components were discovered; uncertainties generated by these factors should be adequately integrated in the input-output model and the state-space model. The uncertainties in the system are primarily caused by three factors: inadequate hydraulic information, simplification of the chlorine kinetics description, and discretization of time-varying chlorine transport delays. The model structural error is obviously deterministic. As a result, it is suggested that the bounding approach be used to model such deterministic structure errors. This structure error is operating point dependant and can be rather high under certain conditions. Because the model will be employed in MPC operations, the full input space must be searched in order to solve the optimal MPC problem. As a result, the a priori knowledge on structure error required in applying the bounding approach for parametric model estimate must include the error structure boundaries corresponding to all potential operating conditions. In the thesis, *EPANET* was used as the DWDS simulator that is an important component in the estimation structure described.

# Chapter 3

# Empirical Modeling

## 3.1 Introduction

Engineers and applied scientists (EASs) frequently find the need for mathematical models to anticipate and project essential performance variables or attributes associated with products or process outcomes. These variables can be broadly categorized into two primary types:

1. **Dependent Variables (Response Variables)**:

   Dependent variables, also known as response variables, are the quantities or characteristics that engineers and scientists are interested in predicting or forecasting. These variables are usually the main focus of the analysis and modeling. They are influenced by one or more independent variables and represent the outcome or result of a process or system. Examples of dependent variables include:

   Temperature in a chemical reaction. Sales revenue of a product. Energy consumption in a manufacturing process. Blood pressure in a medical study.

2. **Independent Variables (Predictor Variables)**:

   Independent variables, also referred to as predictor variables or features, are the factors that are believed to influence or affect the dependent variable. Engineers and scientists collect data on these variables to build mathematical models that can explain and predict the behavior of the dependent variable. Independent variables can be either continuous or categorical. Examples of independent variables include:

   Time: Often used in time series analysis. Temperature: Affects the growth rate of microorganisms. Pressure: Influences the efficiency of a mechanical system. Material composition: Affects the strength of a material. Customer demographics: Influences consumer purchasing behavior. The process of developing a mathematical model involves identifying the relationship between the dependent and independent variables

based on available data. This relationship can be linear or nonlinear, and various techniques such as regression analysis, machine learning algorithms, and neural networks can be employed to create accurate predictive models.

Engineers and applied scientists analyze these mathematical models to gain insights, make informed decisions, optimize processes, and develop strategies for improving performance, efficiency, or quality. The ability to forecast the behavior of crucial performance variables or product/process features is essential for driving innovation, problem-solving, and continuous improvement in engineering and applied science disciplines.

While we will commonly employ various regression techniques to create models based on real data, for clarity, we will term the input variables as "regressors" and the output variables as "responses." This nomenclature helps prevent any potential confusion. The input variables are also labeled as "factors" when addressing a data collection method called factorial experiments. Moreover, the EAS necessitates the utilization of a mathematical equation to accurately define the connections between the input and output variables.

$$Outputs = f(Inputs)$$

Even when the true function "$f$" remains elusive, we possess a means to approximate it. However, this approximation process requires us to gather empirical data across various scenarios. In a positive aspect, a methodology has been formulated to ascertain the minimal conditions required for data collection in order to approximate function "$f$" using a polynomial function. However, this approximation is contingent upon the assumption that function "$f$" is at least piecewise continuous and differentiable.

The encouraging aspect is that most functions we typically encounter exhibit piecewise continuity and differentiability, especially within the input range of interest. Nevertheless, the challenge lies in the fact that "$f$" might exhibit substantial nonlinearity, making it difficult to fit polynomial functions of any degree effectively [37]. In practice, we often resort to straightforward transformations, usually applied to input variables, to approximate "$f$" using polynomials in the transformed variables.

There exist notable and significant special scenarios for the function "$f$". In the initial scenario, either all or a subset of the inputs are discrete or categorical in nature. While certain similar procedures might be applied to estimate "$f$" in these cases, the interpretation would diverge significantly from situations involving continuously valued inputs.

The third case arises when the general structure of "$f$" is known or can be approximated, yet the coefficients or parameters exhibit substantial nonlinearity [37].

## 3.2 Empirical Modeling Principles

Empirical Modeling (EM) constitutes a computer-centered modeling methodology. EM departs from the conventional object-oriented perspective and instead embraces an observation-oriented approach to understanding phenomena. This approach seeks to explain events by focusing on the activities that alter the states of agents. The actions and interactions of agents are translated into modifications in observables. Dependencies are portrayed as the means through which modifications in one observable intricately affect changes in other observables in a unified conceptual manner[37].

## 3.3 Neural Networks Approach

Neural networks belong to a category of machine learning models that draw inspiration from the configuration and operation of the human brain. They excel in tasks that require pattern recognition, classification, regression, and other intricate data-driven operations.

Here's a basic overview of the neural network approach:

1. **Architecture**:

   Neural networks are constructed with interconnected nodes, often referred to as neurons, which are arranged into layers. The prevalent types of layers include:

   - Input Layer: This layer serves as the initial recipient of data or features.
   - Hidden Layers: These layers, which can be multiple, process the data through a series of weighted connections.
   - Output Layer: This layer produces the final prediction or output.

2. **Weights and Activation Functions**:

   Every connection linking neurons is accompanied by a corresponding weight, which undergoes adjustments during the training process. The weighted sum of inputs is then directed through an activation function within each neuron. Standard activation functions encompass sigmoid, tanh, and ReLU (Rectified Linear Unit).

3. **Feedforward Process**:

   In the feedforward process, information traverses from the input layer, traversing through the hidden layers, and ultimately reaching the output layer. Neurons within a given layer are interconnected with neurons in the subsequent layer, with the weighted inputs being subjected to transformation through the activation functions.

4. **Loss Function**:

   A loss function measures the disparity between the predictions generated by the network and the true target values. Throughout the training process, the objective is to minimize this loss by fine-tuning the weights.

5. **Backpropagation**:

   Backpropagation involves the computation of gradients for the loss function in relation to the weights. These gradients provide insights into the extent to which each weight should be modified to decrease the loss.

6. **Optimization**:

   Optimization algorithms, such as Stochastic Gradient Descent (SGD) and its variations (like Adam and RMSProp), employ the gradients derived from backpropagation to adjust the weights within the network. This iterative process enhances the model's performance over time.

7. **Training**:

   The training of the network entails providing it with a dataset and iteratively refining the weights to minimize the loss. This procedure persists until the model reaches a state of convergence where the loss is minimized.

8. **Validation and Testing**:

   Following the training phase, the model is assessed using an independent validation or test dataset to gauge its capacity for generalization. This process validates whether the model can make precise predictions on new, unseen data.

Neural networks find application in diverse domains like image recognition, natural language processing, and speech recognition. Distinct architectures, such as Convolutional Neural Networks (CNNs) tailored for images and Recurrent Neural Networks (RNNs) designed for sequences, have emerged to cater to specific data types and tasks.

However, it's important to emphasize that while neural networks possess significant potential, they demand meticulous hyperparameter tuning, substantial data availability, and adequate computational resources for effective training. Ongoing developments in neural network designs and training methodologies have yielded breakthroughs across domains like artificial intelligence, computer vision, and natural language processing[43].

## 3.4 Fuzzy Clustering

Clustering techniques are unsupervised methodologies employed to arrange data into clusters or groups, guided by similarities among individual data points. Diverging from conventional statistical techniques, clustering algorithms don't hinge on assumptions about the inherent statistical distribution of the data. This quality renders them particularly valuable when there's limited preexisting understanding of the data characteristics [31].

Clustering algorithms have a wide range of applications, including:

-Classification: Clusters can be used to assign new data points to existing groups, aiding in classification tasks.

-Image Processing: Grouping similar regions in images can help with tasks like object recognition and segmentation.

-Pattern Recognition: Clustering can help identify patterns or trends in data that might not be apparent otherwise.

-Modeling: Clusters can serve as a basis for building predictive models.

-Identification: Clustering can assist in identifying groups with specific characteristics within a dataset.

Clustering algorithms can handle various types of data, including:

-Quantitative (Numerical) Data: This refers to data that can be measured using numbers. For example, measurements of length, temperature, etc.

-Qualitative (Categorical) Data: This involves non-numeric data that represents categories or labels. For example, colors, types of animals, etc.

-Mixed Data: Data that includes a combination of both numerical and categorical features.

The chapter primarily focuses on clustering quantitative data. This means that the data used in the clustering process consist of numerical measurements.

Every entry within the dataset comprises "$n$" measurable variables. These variables are structured into an "$n$" dimensional column vector denoted as $Z_k = [z_{1k}, ..., z_{nk}]$, $Z_k \in R^n$. A collection of $N$ observations is represented as

$Z = \{z_k \,|k = 1, 2, ..., N\}$ and is represented as an $n * N$ matrix.

$$Z = \begin{bmatrix} z_{11} & z_{12} & \cdots & z_{1N} \\ z_{21} & z_{22} & \cdots & z_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ z_{n1} & z_{n2} & \cdots & z_{nN} \end{bmatrix} \tag{3.1}$$

The matrix "Z" is a fundamental representation of the data used in clustering and pattern recognition. It's a structured way to organize the data for analysis. In this context, each row of the matrix corresponds to a data point (pattern or object), and each column corresponds to a specific attribute or feature of the data. In the realm of pattern recognition, the columns of matrix "Z" are referred to as "patterns" or "objects." These patterns could represent individual instances, samples, or observations from a dataset. The rows of matrix "Z" are described as "features" or "characteristics." Each row holds the values of different attributes or variables that describe the corresponding pattern or object. These attributes could be measurements, properties, or qualities associated with each pattern. For example, In the context of medical diagnosis, if each column of "Z" represents individuals (patients), then each row could represent symptoms or laboratory measurements associated with these patients. This creates a matrix where you have different symptoms or lab values for each patient.If clustering is applied to study dynamic systems, the columns of "Z" might contain temporal signal samples. Each column could represent a time step, and each row could contain physical variables (position, pressure, temperature) observed at that particular time step. This way, you can analyze the system's behavior and relationships over time. In dynamic systems, past values of variables are often included in the matrix "Z." This incorporation reflects the system's dynamics and allows for the consideration of historical information [3].

Depending on the specific objective of clustering, different definitions of a cluster can be formulated. Generally, a cluster is understood as a group of items that exhibit more similarity to each other than to members of other clusters. This concept of "similarity" is based on mathematical measures and is quantified using well-defined methods. Similarity in metric spaces is typically defined by a distance norm. The distance between data vectors or between data vectors and a prototypical object (prototype) of the cluster can be measured. Clustering algorithms are used in conjunction with data partitioning to look for prototypes that are often unknown in advance. Vectors of the same dimension as data items can be prototypes, but they can additionally represent higher-level geometrical structures such as linear or nonlinear subspaces or functions.

Data can reveal clusters characterized by diverse geometric forms, dimensions, and concentrations, as visually depicted in Figure 3.1. Cluster arrangement (a) displays spherical geometry, while clusters (b) through (d) showcase linear and nonlinear substructures within the data space [3].

The efficacy of most clustering methods is influenced by factors such as the geometric shapes and densities of individual clusters, along with the spatial interactions and distances between these clusters. Clusters can exhibit characteristics of being well-separated, consistently connected, or overlapping[1].

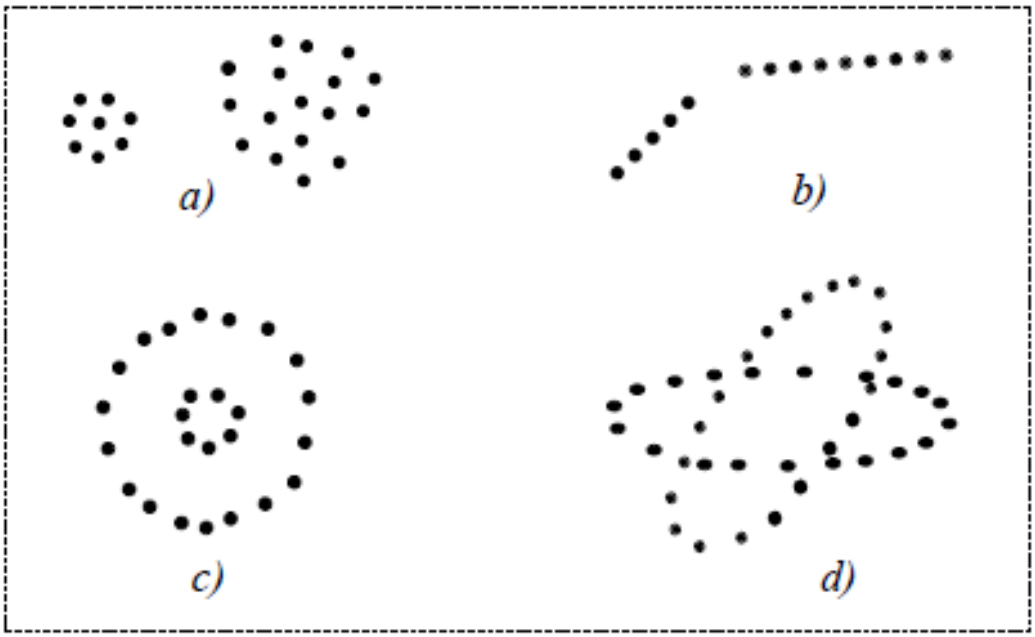


Fig. 3.1 Clusters of different shapes and dimensions

Clustering is a fundamental technique in data analysis and pattern recognition, and it involves grouping similar data points together to form clusters. These clusters can help in uncovering hidden structures, patterns, and relationships within a dataset. As you mentioned, one way to categorize clustering methods is based on whether the resulting clusters are fuzzy or crisp (hard)[49].

Hard clustering, also known as exclusive clustering, is where each data point is assigned to exactly one cluster. In this approach, there is a clear boundary between clusters, and each data point belongs to the cluster it is most similar to according to a certain distance or similarity measure.

Fuzzy clustering, in fact, provides greater adaptability in grasping the intrinsic uncertainty and overlapping characteristics frequently observed in real-world datasets. This is achieved by enabling data points to be associated with multiple clusters simultaneously, each with differing degrees of membership. Through this mechanism, fuzzy clustering can offer a more intricate depiction of intricate data interconnections.

The agglomerative and divisive hierarchical methods create new clusters by redistributing the memberships of individual points, utilizing a relevant similarity measure. Alternatively, graph-theoretic techniques treat the collection Z as a set of nodes, establishing edge

weights according to similarity measures between these nodes. Clustering algorithms often employ an objective function to evaluate the partition quality.

Nonlinear optimization algorithms are employed to identify the local optima of the objective function. The subsequent sections of this chapter are dedicated to exploring fuzzy clustering with a designated target function. These methodologies are reasonably comprehensible, and mathematical solutions exist to ascertain convergence characteristics and cluster validity.

The notion of fuzzy partitioning holds pivotal importance in cluster analysis and, by extension, in identification methods grounded in fuzzy clustering. The conventional hard partitions, delineated through classical subsets, are expanded upon by incorporating fuzzy and probabilistic partitions [52].

### 3.4.1 Hard Clustering

Hard clustering, often referred to as crisp clustering, is an unsupervised machine learning approach utilized to segregate a dataset into discrete and non-overlapping clusters. In this method, each data point is associated with precisely one cluster, leading to distinct and unambiguous assignments. This implies that cluster assignment decisions are definitive, devoid of overlap or uncertainty in the assignment of data points to clusters [14].

**Key Characteristics**:

- Exclusive Membership: Within hard clustering, every individual data point is attributed to a sole cluster, and there's no provision for inclusion in any other cluster.
- Clear Separation: The primary goal of hard clustering is to establish clusters that exhibit clear separation, ensuring that data points within the same cluster demonstrate greater similarity among themselves compared to data points belonging to different clusters.
- Fixed Cluster Number: Most hard clustering algorithms require you to specify the number of clusters you want to create beforehand.

**Advantages**:

- Simplicity: Hard clustering methods are often simpler to implement and understand compared to fuzzy clustering or probabilistic models.
- Interpretability: The resulting clusters have clear boundaries, which makes them easy to interpret and explain.

- Efficiency: Many hard clustering algorithms are computationally efficient, making them suitable for large datasets.

**Disadvantages**:

- Lack of Flexibility: Hard clustering doesn't handle cases where data points could reasonably belong to multiple clusters well, or when there's uncertainty about cluster assignments.
- Sensitive to Noise: Hard clustering can be sensitive to outliers and noise, which might lead to inaccurate cluster assignments.
- Need for Prior Knowledge: You often need to know the number of clusters in advance, which might not always be the case in real-world scenarios.

Hard clustering is suitable when your data naturally falls into distinct, non-overlapping groups and when you have a good understanding of the number of clusters you want to create. However, it might struggle when there's uncertainty in cluster assignments or when data points lie on the boundaries between clusters. In such cases, fuzzy clustering or other more flexible techniques might be more appropriate[14].

It's worth highlighting that hard partitioning encounters challenges when confronted with intricate or overlapping patterns within the data. In such scenarios, fuzzy or possibilistic clustering offers greater flexibility by accommodating partial memberships or degrees of possibility. This leads to a more adaptable and precise depiction of the underlying data structures.

To summarize, the process of partitioning a dataset Z into c clusters encompasses the exclusive assignment of each data point to a solitary cluster within hard partitioning. The objective is to craft well-defined and internally cohesive clusters, grounded in a predefined similarity metric [14].

$\{A_i \,|1 \leq i \leq c\} \subset \rho \mathrm{P}(Z)$ with the following properties

where $P(Z)$ is the power set of Z.

$$\bigcup_{i=1}^{c} A_i = Z \tag{3.2a}$$

$$A_i \cap A_j = \emptyset, 1 \leq i \neq j \leq c \tag{3.2b}$$

$$\emptyset \subset A_i \subset Z, 1 \leq i \leq c \tag{3.2c}$$

The combined subsets $A_i$, as defined by Equation (3.2a), encompass the entire dataset. As stipulated by Equation (3.2b), these subsets are required to be non-overlapping, with

none of them being empty or encompassing the entirety of data points in Z as outlined in Equation (3.2c).

A partition can be conveniently articulated using *membership* (*characteristic*) *functions* through the utilization of the *partition matrix* $U = [\mu_{ik}]_{c\times N}$. Each row of this matrix, denoted by $\mu_i$, encapsulates values representing the membership function of Z's *ith* subset $A_i$. In accordance with Equation (3.2), the matrix elements within U must adhere to these criteria:

$$\mu_{ik} \in \{0,1\}, 1 \leq i \leq c, 1 \leq k \leq N \tag{3.3a}$$

$$\sum_{i=1}^{c} \mu_{ik} = 1, 1 \leq k \leq N \tag{3.3b}$$

$$0 < \sum_{k=1}^{N} \mu_{ik} < N, 1 \leq i \leq c \tag{3.3c}$$

The hard partitioning space is thus defined as the set of all possible hard partition matrices for Z [52] :

$$M_{hc} = \left\{ U \in \Re^{c\times N} \middle| \mu_{ik} \in \{0,1\}, \forall i,k; \sum_{i=1}^{c} \mu_{ik} = 1, \forall k; 0 < \sum_{k=1}^{N} \mu_{ik} < N, \forall i \right\}$$

- **Example**:

Let us use a simple example to demonstrate the notion of hard partitioning [3]. Consider the data set $Z = \{z_1, z_2, ...z_{10}\}$, as given in Figure3.2

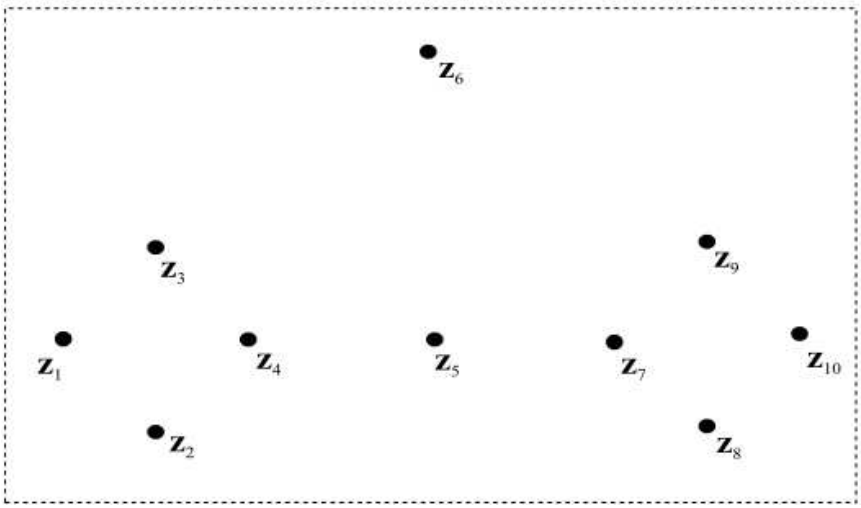


Fig. 3.2 A data set in $\Re^2$

$$U = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

The first row of matrix U represents the characteristic function corresponding to the initial subset of Z, denoted as $A_1$, while the subsequent rows define characteristic functions for subsequent subsets like $A_2$. Each individual sample is required to be uniquely allocated to a single partition subset or cluster. In this specific scenario, both the border point $z_5$ and the outlier $z_6$ have been assigned to the subset $A_1$. Evidently, this instance highlights that hard partitioning might not accurately portray the underlying data. Instances situated at the boundary can exhibit patterns that encompass a mixture of data attributes from both $A_1$ and $A_2$, and as a result, they cannot be fully attributed to either of these classes, nor form a distinct class.

This limitation can be addressed by leveraging fuzzy and probabilistic partitions, as elucidated in subsequent sections [3].

### 3.4.2 Fuzzy Partition

Fuzzy partitioning is a fundamental concept within cluster analysis and plays a pivotal role in fuzzy clustering methodologies. It facilitates the assignment of membership degrees to individual data points, indicating the extent to which each point belongs to various clusters.

Unlike hard partitioning, where each data point is exclusively assigned to a single cluster, fuzzy partitioning permits overlapping memberships. This implies that data points can possess partial memberships in multiple clusters, reflecting the level of uncertainty or ambiguity in their assignments.

Fuzzy clustering techniques, including the Fuzzy C-Means (FCM) algorithm, leverage fuzzy partitions to gauge the degree of membership for data points within each cluster. By accounting for fuzzy memberships, these algorithms can capture the inherent fuzziness or ambiguity present in the data, thereby yielding more versatile and nuanced clustering outcomes.

Likewise, possibilistic clustering methodologies expand on the concept of hard partitioning by allowing for partial or possibilistic memberships. Possibilistic clustering assigns a degree of possibility to each data point's membership within a cluster, indicating the extent to which the point might belong to that particular cluster.

Fuzzy and possibilistic partitions can be perceived as extensions of hard partitions, framed within classical subsets. They establish a more adaptable framework for the exploration of clustering, enabling a finer-grained representation of the underlying data structures and associated uncertainties.

These concepts of fuzzy and possibilistic partitions have demonstrated successful application across diverse fields, encompassing pattern recognition, image processing, and data mining. Particularly in scenarios where data exhibits complex and overlapping patterns, these approaches enhance the interpretability and accuracy of clustering outcomes by introducing the notion of fuzzy or possibilistic memberships.

The hard partition is directly generalized to the fuzzy case by allowing $\mu_{ik}$ to acquire real values in [0,1], [45] provides conditions for a fuzzy partition matrix that are equivalent to(3.2).

$$\mu_{ik} \in [0,1], 1 \leq i \leq c, 1 \leq k \leq N \tag{3.4a}$$

$$\sum_{i=1}^{c} \mu_{ik} = 1, 1 \leq k \leq N \tag{3.4b}$$

$$0 < \sum_{k=1}^{N} \mu_{ik} < N, 1 \leq i \leq c \tag{3.4c}$$

The values of the ith membership function of the fuzzy subset $A_i$ of Z are contained in the $i_{th}$ row of the fuzzy partition matrix U. Equation(3.4b) limits the sum of each column to one, therefore the total membership of each $z_k$ in Z is one. The set is the fuzzy partitioning space for Z[4].

$$M_{fc} = \left\{ U \in \Re^{c \times N} \middle| \mu_{ik} \in [0,1], \forall i,k; \sum_{i=1}^{c} \mu_{ik} = 1, \forall k; 0 < \sum_{k=1}^{N} \mu_{ik} < N, \forall i \right\}$$

- **Example**:

Consider the data set from the previous. One of the infinitely many fuzzy partitions in Z is:

$$U = \begin{bmatrix} 1.0 & 1.0 & 1.0 & 0.8 & 0.5 & 0.5 & 0.2 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.2 & 0.5 & 0.5 & 0.8 & 1.0 & 1.0 & 1.0 \end{bmatrix}$$

The border point $z_5$ now exhibits a membership degree of 0.5 in both classes, reflecting its placement at the midpoint between the two clusters. Conversely, the outlier $z_6$ demonstrates matching pairs of membership degrees in both clusters, despite being farther away. As a result, it can be considered less representative of both $A_1$ and $A_2$ in comparison to $z_5$. This is due to the requirement outlined in equation(3.4b), which necessitates that the sum of membership degrees for each point totals one.

In this context, employing three clusters instead of two would provide a more accurate representation. Generally, the identification and assignment of outliers to additional clusters

pose challenges. This drawback of fuzzy partitions is circumvented through the application of possibilistic partition, elucidated in the subsequent section[46].

### 3.4.3 Possibilistic Clustering

Relaxing the constraint established by Equation (3.4b) results in a more inclusive variant of fuzzy partitioning, referred to as possibilistic clustering. Nevertheless, it's essential to acknowledge that completely removing this constraint is not feasible in order to ensure that every data point is allocated to a fuzzy subset with a membership value greater than zero.

Instead of Equation(3.4b), a less stringent constraint can be adopted: $\forall k$, $\exists i$, and $\mu_{ik} > 0$. The prerequisites for a possibilistic fuzzy partition matrix are delineated as follows[4]:

$$\mu_{ik} \in [0,1], 1 \leq i \leq c, 1 \leq k \leq N \tag{3.5a}$$

$$\exists i, \mu_{ik} > 0, \forall k \tag{3.5b}$$

$$0 < \sum_{k=1}^{N} \mu_{ik} < N, 1 \leq i \leq c \tag{3.5c}$$

The set is the possibilistic partitioning space for Z, as in the prior situations

$$M_{pc} = \left\{ U \in \Re^{c \times N} \,|\, \mu_{ik} \in [0,1], \forall i,k \,; \forall k, \exists i, \mu_{ik} > 0; 0 < \sum_{k=1}^{N} \mu_{ik} < N, \forall_i \right\}$$

- **Example**:

For our data collection, below is an example of a possibilistic partition matrix:

$$U = \begin{bmatrix} 1.0 & 1.0 & 1.0 & 1.0 & 0.5 & 0.2 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.5 & 0.2 & 1.0 & 1.0 & 1.0 & 1.0 \end{bmatrix}$$

Because the sum of the elements in each column of $U \subset M_{pc}$ is no longer constrained, the outlier has a membership of 0.2 in both clusters, which is less than the membership of the border point $z_5$, indicating that this point is less typical for the two clusters than $z_5$.

### 3.4.4 Fuzzy C-means Clustering

The Fuzzy C-Means (FCM) algorithm is a clustering technique that permits a singular data point to have membership in two or more clusters.

**The Fuzzy c-Means Functional**

A wide range of fuzzy clustering methodologies centers around minimizing the fuzzy C-Means functional, which is mathematically defined as [4]:

$$J(Z;U;V) = \sum_{i=1}^{c}\sum_{k=1}^{N}(\mu_{ik})^m \|z_k - v_i\|_A^2 \tag{3.6a}$$

where

$$U = [\mu_{ik}] \in M_{fc} \tag{3.6b}$$

U is defined as the fuzzy partition matrix of Z,

$$V = [v_1, v_2, ..., v_c], v_i \in \Re^n \tag{3.6c}$$

V denotes a vector of cluster prototypes (centers) that must be defined.

$$D_{ikA}^2 = \|z_k - v_i\|_A^2 = (z_k - v_i)^T A (z_k - v_i) \tag{3.6d}$$

is a squared inner-product distance norm, and

$$m \in [1, \infty) \tag{3.6e}$$

signifies the factor that dictates the degree of fuzziness in the resultant clusters. The cost function's value, as indicated by Equation(3.6a), can be regarded as a gauge of the overall variance between $z_k$ and $v_i$.

**The Fuzzy c-Means Algorithm**

The c-means functional minimization requires addressing a nonlinear optimization issue, which can be approached using a variety of methods, including iterative minimization. These methods provide approximations to issue solutions and are often utilized in clustering algorithms such as FCM.The fuzzy c-means (FCM) technique is based on a simple Picard iteration over the first-order conditions for stationary points in (3.6a). The stationary points of the objective function(3.6a) can be obtained by adjoining the constraint(3.6b) to J by means of Lagrange multipliers[4]:

$$\bar{J}(Z;U,V,\lambda) = \sum_{i=1}^{c}\sum_{k=1}^{N}(\mu_{ik})^m D_{ikA}^2 + \sum_{k=1}^{N}\lambda_k\left[\sum_{i=1}^{c}\mu_{ik} - 1\right] \tag{3.7}$$

and by setting the gradients of $\bar{J}$ with respect U, V and $\lambda$ to zero. it can be shown that if $D_{ikA}^2 > 0$, $\forall i$, k and $m > 1$, then $(U,V) \in M_{fc} \times \Re^{n\times c}$ may minimize(3.6a) only if

$$\mu_{ik} = \frac{1}{\sum_{j=1}^{c} \left( D_{ikA} / D_{jkA} \right)^{2/(m-1)}}, 1 \leq i \leq c, 1 \leq k \leq N, \tag{3.8a}$$

and

$$v_i = \frac{\sum_{k=1}^{N} (\mu_{ik})^{m} z_k}{\sum_{k=1}^{N} (\mu_{ik})^{m}}; 1 \leq i \leq c. \tag{3.8b}$$

This solution also meets the other constraints(3.4a)and (3.4c). The(3.8) equations are first-order necessary conditions for stationary points of the functional (3.6a). Iteratively, the FCM algorithm goes through(3.8a)and (3.8b) (Bezdek, 1980) establishes the sufficiency of (3.8)and the convergence of the FCM algorithm. It's important to highlight that (3.8b) calculates $v_i$ as the weighted average of data items associated with a cluster, with membership degrees serving as the weights. This is the rationale behind the method's name, known as c-means [4].

Choose the number of clusters 1 < c < N, the weighting exponent m > 1, the termination tolerance $\varepsilon > 0$, and the norm-inducing matrix A from the data set **Z**. Randomly initialize the partition matrix so that $U(0) in M_f c$

**Repeat for** l=1,2,...

**Step 1** : Compute the cluster prototypes (means)

$$v_i^{(l)} = \frac{\sum_{k=1}^{N} \left( \mu_{ik}^{(l-1)} \right)^{m} z_k}{\sum_{k=1}^{N} \left( \mu_{ik}^{(l-1)} \right)^{m}}, 1 \leq i \leq c$$

**Step 2**: Compute the distances

$$D_{ikA}^2 = \left( z_k - v_i^{(l)} \right)^T A \left( z_k - v_i^{(l)} \right), 1 \leq i \leq c, 1 \leq k \leq N$$

**Step 3**: Update the partition matrix
for $1 \leq k \leq N$
if $D_{ikA} >$ for all i=1,2,...,c

$$\mu_{ik}^{(l)} = \frac{1}{\sum_{j=1}^{c} \left( D_{ikA} / D_{jkA} \right)^{2/(m-1)}}$$

otherwise
$\mu_{ik}^{l}$=0 if $D_{ikA}$>0, and $\mu_{ik}^{l} \in [0,1]$ with $\sum_{i=1}^{c} \mu_{ik}^{(l)} = 1$

**until** $\left\| U^{(l)} - U^{(l-1)} \right| < \varepsilon$

for which $D_{ikA} > 0$ and the memberships are distributed arbitrarily among the clusters for which $D_{ikA} = 0$, such that the constraint in(3.4b) is satisfied.

#### Parameters of the FCM Algorithm

Several parameters must be set before using the Fuzzy C-Means (FCM) algorithm: the number of clusters, c, the fuzziness exponent, m, the termination tolerance, $\varepsilon$, and the norm-inducing matrix, A. The fuzzy partition matrix, U, must also be initialized. These parameters' choices are now described one by one[4].

**Number of Clusters**. The number of clusters, denoted as c, is indeed a crucial parameter in clustering algorithms. It determines the desired number of clusters in the data and has a significant impact on the resulting partition. The choice of the number of clusters affects the granularity and organization of the clustering solution. When the number of clusters is set correctly and matches the underlying structure of the data, the clustering algorithm has a higher chance of identifying and separating the distinct groups effectively. In such cases, the resulting partition is more likely to align with the true clusters in the data. When clustering real data without any prior information about the underlying structures, it is often necessary to make assumptions about the number of clusters. These assumptions help guide the clustering algorithm in searching for a specific number, denoted as c, of clusters in the data.

However, it's important to note that the chosen number of clusters may not necessarily correspond to the true or optimal number of clusters present in the data. In some cases, the true number of clusters may not be clearly distinguishable or may not match the assumed number.

The clustering algorithm will attempt to partition the data into c clusters based on the given assumption. It will analyze the patterns and relationships in the data to create the desired number of clusters. The algorithm may use different techniques, such as optimizing an objective function, maximizing cluster separation, or minimizing within-cluster variance[14].

It is important to be aware that the assumption about the number of clusters is often a subjective decision made by the user or based on prior knowledge or hypotheses. Therefore, the results obtained from the clustering algorithm should be carefully evaluated and validated to assess the appropriateness of the chosen number of clusters.

In practice, selecting the correct number of clusters can be challenging, and it often requires iterative experimentation and evaluation. Various techniques, such as validation indices, visualization methods, or domain knowledge, can be employed to determine the optimal number of clusters or to assess the quality of the clustering solution.

Ultimately, the selection of the number of clusters is an important consideration in clustering analysis, and it should be based on careful consideration of the data characteristics, the goals of the analysis, and an understanding of the limitations of the chosen clustering algorithm.

1. Validity measures.

Validity measures in clustering refer to scalar indices that assess the quality or goodness of a obtained partition. These measures are used to evaluate how well a clustering algorithm performs in terms of locating well-separated and compact clusters.

Ideally, in a clustering task, the goal is to identify clusters that correspond to the underlying groups or patterns in the data. If the number of clusters chosen for the algorithm is equal to the actual number of groups present in the data, it is reasonable to expect that the clustering algorithm will be able to identify them correctly.

When the number of clusters matches the true number of groups, it increases the likelihood of obtaining a good partition. In such cases, the clustering algorithm has a higher chance of correctly separating the different groups and assigning the data points to their respective clusters.

However, it's important to note that the success of clustering algorithms is not solely determined by the number of clusters chosen. Other factors such as the choice of algorithm, the quality of the data, and the suitability of the clustering technique for the specific data distribution also play crucial roles in obtaining accurate and meaningful clusters.

Validity measures can provide quantitative assessments of the quality of the clustering results, such as compactness and separation of clusters. These measures help in evaluating the performance of different clustering algorithms and guiding the selection of appropriate clustering techniques for a given dataset. When the number of clusters chosen does not match the actual number of groups in the data, misclassifications may occur, and the resulting clusters are less likely to be well separated and compact. Therefore, cluster validity measures are designed to quantify the separation and compactness of the clusters[14].

One popular validity measure for the Fuzzy C-Means (FCM) algorithm is the Xie-Beni index, introduced by Xie and Beni in 1991. The Xie-Beni index evaluates the compactness and separation of clusters based on the objective function value of the FCM algorithm.

The Xie-Beni index, denoted as c (in the original notation), is a validity measure that aims to find a balance between the compactness of the clusters (minimizing intra-cluster variance)

and the separation between clusters (maximizing inter-cluster distances). It provides a single scalar value that quantifies the quality of the obtained clustering solution.

The specific formula and computation of the Xie-Beni index may vary depending on the study or implementation. However, the general idea is to compute the ratio of the sum of the within-cluster variances to the separation between clusters. Lower values of the Xie-Beni index indicate better cluster validity, indicating more compact and well-separated clusters.

Overall, validity measures, including the Xie-Beni index, help in assessing the quality of clustering results and can guide the selection of suitable clustering algorithms or parameter settings for a given dataset.

$$\chi(Z;U,V) = \frac{\sum_{i=1}^{c}\sum_{k=1}^{N} \mu_{ik}^{m} \|z_k - v_i\|^2}{c.\min_{i\neq j}\left(\left\|v_i - v_j\right\|^2\right)} \tag{3.9}$$

In practice, it has proven to be effective. This index can be concidered as the ratio of the total within-group variance and the separation of the cluster centers. The best partition minimizes the value of $\chi(Z;U,V)$ [26].

1. Iterative merging or insertion of clusters

Cluster merging is a technique used in clustering algorithms to minimize the number of clusters by iteratively combining similar or compatible clusters based on certain criteria. The objective is to create a more compact and meaningful interpretation of the data by grouping similar clusters together.

The fundamental concept behind cluster fusion is to begin with a sufficiently large number of initial clusters. These initial clusters can be generated using a clustering algorithm or any other method. Then, through successive iterations, the algorithm evaluates the similarity or compatibility between pairs of clusters based on defined criteria.

The criteria for merging clusters can vary depending on the specific clustering algorithm or application. Commonly used criteria include measures of distance or similarity between cluster centroids, overlap of cluster boundaries, or statistical properties of the data within the clusters.

The merging process typically involves selecting the pair of clusters with the highest similarity or compatibility and combining them into a single cluster. This process continues iteratively until a stopping condition is met, such as reaching a desired number of clusters or when no more merges can be performed.

on the other side, clustering algorithms can start with a small number of clusters and iteratively insert new clusters into regions where data points have a low degree of membership in the current clusters. This method is referred to as cluster splitting or cluster insertion. Its

goal is to find finer-grained clusters inside regions that were previously assigned to larger clusters[12].

Both cluster merging and cluster splitting strategies can be useful in adapting the clustering process to the characteristics of the data and improving the overall quality of the clustering solution. The choice of the approach depends on the specific requirements of the problem and the nature of the data being analyzed[26].

**Fuzziness Parameter**. The weighting exponent $m$ is also a crucial parameter because it determines the fuzziness of the final partition. The partition becomes hard as $m$ approaches one from above. $\mu_{ik} \in (0,1)$ and $v_i$ are typical cluster means. The partition becomes entirely fuzzy as $m \longrightarrow \infty$, $\mu_{ik} = 1/c$, and the cluster means are all equivalent to the mean of Z. These (3.6) limit features are independent of the optimization approach utilized. m = 2 is usually selected at first [3].

**Termination Criterion**. When the norm of the difference in U between two consecutive iterations is less than the termination parameter $\varepsilon$ For the maximum norm, $max_{ik}\left(\left|\mu_{ik}^{(l)} - \mu_{ik}^{(l-1)}\right|\right)$ The FCM algorithm has finished iterating, Usually, $\varepsilon = 0.001$, even though $\varepsilon = 0.01$ performs well, while radically decreasing the computing times.

**Norm-Inducing Matrix**. In the distance measure (3.6d), which is commonly used in clustering algorithms, the matrix A is used to weight the individual dimensions of the data. When A is set to the identity matrix, each dimension is equally weighted, and the resulting distance measure is the standard Euclidean distance.

The Euclidean norm, induced by the choice of **A = I**, calculates distances between points as the straight-line or Euclidean distance. This distance measure assumes that each dimension contributes equally to the overall dissimilarity between points. As a result, clusters formed using the Euclidean norm tend to have hyperspherical shapes, where surfaces of constant membership are hyperspheres.

$$D_{ik}^2 = (z_k - v_i)^T (z_k - v_i) \tag{3.10a}$$

Another option for A is a diagonal matrix that provides for different variances in the directions of the coordinate axes of Z:

$$A = \begin{bmatrix} (1/\sigma_1)^2 & 0 & \cdots & 0 \\ 0 & (1/\sigma_2)^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & (1/\sigma_n)^2 \end{bmatrix} \tag{3.10b}$$

This matrix establishes a diagonal norm on $\Re^n$. Ultimately, the matrix **A** is the inverse of the covariance matrix of Z, denoted as $Z : A = R^{-1}$ , with

$$R = \frac{1}{N}\sum_{k=1}^{N}(z_k - \bar{z})(z_k - \bar{z})^T \tag{3.10c}$$

Here, the symbol $\bar{z}$ represents the data's mean. In this context, matrix A induces the Mahalanobis norm on $\Re^n$.

The influence of the norm on clustering criteria is observed through modifications in the dissimilarity measurement. The choice of norm affects the shape of clusters in the feature space. With the Euclidean norm, clusters take on a hyperspherical form, with surfaces of consistent membership resembling hyperspheres. On the other hand, both the diagonal and Mahalanobis norms lead to hyperellipsoidal clusters.

In the case of the diagonal norm, the axes of the hyperellipsoid align with the coordinate axes. The Mahalanobis norm, however, permits arbitrary orientations of hyperellipsoids. The orientation is determined by the covariance matrix of the data. Incorporating correlations between distinct features allows clusters to freely orient themselves in the feature space. A visual depiction of this phenomenon can be seen in Figure 3.3.

A shared constraint of clustering algorithms relying on a fixed distance norm is their predisposition to favor clusters of specific shapes, even if such shapes don't correspond to the actual data distribution.

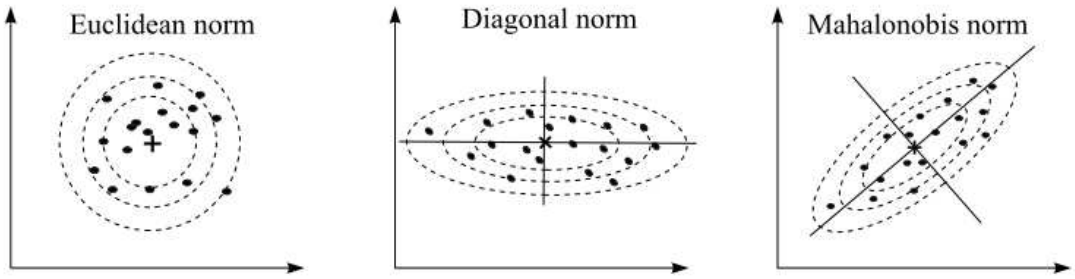


Fig. 3.3 Different Distance Norms used in Fuzzy Clustering

- **Example**: Fuzzy c-means clustering

The given scenario describes a synthetic dataset in $\Re^2$ consisting of two separated clusters with different shapes. Both clusters are generated by sampling from the normal distribution. The upper cluster has a standard deviation of 0.2 for both axes, while the lower cluster has a standard deviation of 0.2 for the horizontal axis and 0.05 for the vertical axis.

The FCM (Fuzzy C-Means) algorithm is applied to this dataset. In FCM, the norm-inducing matrix A is set to the identity matrix (I) for both clusters. The weighting exponent $m$ is set to 2, which indicates that the algorithm aims to produce fuzzy partitions with crisp membership values. The termination criterion is set to $\varepsilon = 0.01$, meaning that the algorithm stops when the change in the objective function falls below this threshold.

The FCM algorithm commences by initializing a random partition matrix and subsequently iterates to refine membership values and cluster centers until reaching a point of convergence. In this instance, convergence is attained within 4 iterations.

Nevertheless, notwithstanding the achieved convergence, the contours of membership levels depicted in Figure 3.4 reveal a constraint inherent in the FCM algorithm. Specifically, the algorithm tends to impose a circular configuration on both clusters, even when the lower cluster is elongated in reality. This limitation stems from the FCM algorithm's presumption that clusters are delineated by circular distributions. Consequently, the algorithm can encounter challenges when striving to effectively represent clusters with non-circular shapes, as exemplified by the elongated lower cluster in this particular scenario.

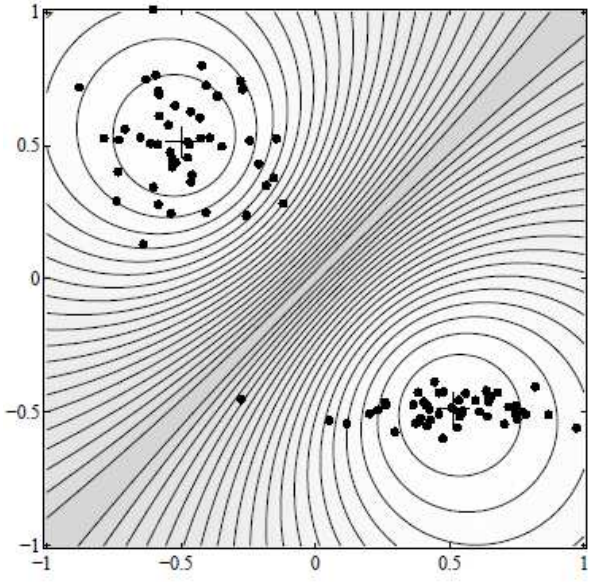


Fig. 3.4 The fuzzy c-means clusters, .. are the data points, + are the cluster means

**Initial Partition Matrix**. The initialization of the partition matrix typically involves a random procedure, ensuring that $U$ conforms to $M_{fc}$. A straightforward method to achieve this entails initializing the cluster centers $v_i$ randomly and subsequently deriving the corresponding matrix $U$ through (3.8a) (essentially, through the utilization of the third step within the FCM algorithm)..

## 3.5 MIMO FUZZY MODEL

### 3.5.1 Takagi-Sugeno Fuzzy Modeling for multivariable nonlinear system

The fuzzy model is constructed based on data acquired from input-output pairs through a tailored model identification algorithm. This algorithm incorporates fuzzy clustering to ascertain the antecedents of the fuzzy model rules, while the consequent segments are established using a least squares parameter estimation technique.

We address a Drinking Water Distribution System (DWDS) functioning as a nonlinear Multiple-Input Multiple-Output (MIMO) system denoted as $\mathscr{G}(U,Y)$. This MIMO system is characterized as an assembly of interconnected discrete-time fuzzy models that adhere to the nonlinear autoregressive model with exogenous input[5].

$$y_\ell(k+1) = R_\ell(\psi_\ell, u(k)), \ell = 1,2,...,p \tag{3.11}$$

In this context, the input vector is represented as $u \in \Re^m$, the output vector as $y \in \Re^p$, and the regression vector $\psi(k) \subseteq \Re^s$ encompasses both the present and prior outputs, as well as historical inputs.

$$u(k) = [u_1(k), u_2(k), ..., u_m(k)]^T \tag{3.12}$$

$$\psi_\ell(k) = [y_1(k), ..., y_p(k), u_1(k-1), ...., u_m(k-1)]^T \tag{3.13}$$

with

$$y_i(k) = [y_i(k), y_i(k-1), ..., y_i(k-n_{y,i})]$$

$$u_j(k-1) = \left[u_j(k-1), u_j(k-2), ..., u_j\left(k-n_{u,j}\right)\right]$$

where $i = 1,...,p$, $j = 1,...,m$, $n_{y,i}$ is the order of the $i^{th}$ output and $n_{u,j}$ is the order of the $j^{th}$ input. .

$$s = \sum_{i=1}^{p}(n_{y,i}+1) + \sum_{j=1}^{m} n_{u,j}$$

corresponds to the count of antecedent variables, whereas $c$ signifies the quantity of clusters, which concurrently denotes the number of fuzzy rules associated with the $\ell^{th}$ output. The notation $\Re_{\ell i}$ designates the T-S type fuzzy models established according to rules [36].

$\Re_{\ell i}$: IF $\psi_{\ell 1}(k)$ is $Z_{\ell i,1}$,...,$\psi_{\ell s}(k)$ is $Z_{\ell i,s}$,$u_1(k)$ is $Z_{\ell i,s+1}$,...,$u_m(k)$ is $Z_{\ell i,s+m}$, THEN

$$y_{\ell i}(k+1) = \mu_{\ell i}\psi(k) + \eta_{\ell i}u(k) + \varphi_{\ell i} \tag{3.14}$$

$i = 1,2,...,c$ ; $\ell = 1,2,...,p$

Here, $Z_{\ell i}$ signifies the antecedent fuzzy set corresponding to the $\ell^{th}$ output and the $i^{th}$ rule. The vectors $\mu_{\ell i}$ and $\eta_{\ell i}$ encapsulate the resulting parameters, while $\varphi_{\ell i}$ represents the offset term. The model's output is computed as the weighted mean of the linear consequents present within each distinct rule.

$$y_\ell(k+1) = \frac{\sum_{i=1}^{c} \beta_{\ell i}(\psi_\ell(k), u(k))(\mu_{\ell i}\psi(k) + \eta_{\ell i}u(k) + \varphi_{\ell i})}{\sum_{i=1}^{c} \beta_{\ell i}(\psi_l(k), u(k))} \tag{3.15}$$

with

$$\beta_{li}(\psi_l(k), u(k)) = \prod_{h=1}^{s} \lambda_{Z_{\ell i,h}}(\psi_{\ell h}) \prod_{j=1}^{m} \lambda_{Z_{\ell i,j}}(u_j)$$

In this context, $\lambda_{Z_{\ell i}}(.)$ represents the membership degrees associated with the antecedent variables, and $\Pi$ stands for the fuzzy operator employed.

Let

$$\bar{\beta}_{\ell i}(\psi_\ell(k), u(k)) = \frac{\beta_{\ell i}(\psi_\ell(k), u(k))}{\sum_{i=1}^{c} \beta_{\ell i}(\psi_\ell(k), u(k))} \tag{3.16}$$

with

$$\sum_{i=1}^{c} \bar{\beta}_{\ell i}(\psi_\ell(k), u(k)) = 1, 0 \leq \bar{\beta}_{\ell i}(\psi_\ell(k), u(k)) < 1 \tag{3.17}$$

Then the $\ell^{th}$ output of the fuzzy models can be written as

$$y_\ell(k+1) = \sum_{i=1}^{c} \bar{\beta}_{\ell i}(\psi_\ell(k), u(k))(\mu_{\ell i}\psi_\ell(k) + \eta_{\ell i}u(k) + \varphi_{\ell i}) \tag{3.18}$$

## 3.6 T-S FUZZY MODEL IDENTIFICATION

To establish the consequent parameters of T-S models, formulate the following set of equations

$$\begin{aligned} \Psi(k-1) = [&y_1(k-1), \ldots, y_1\left(k-n_{y,1}-1\right), \ldots, y_p(k-1), \ldots, \\ &y_p(k-n_{y,p}-1), u_1(k-2), \ldots, u_1(k-n_{u,1}-1), \ldots, \\ &u_m(k-2), \ldots, u_m(k-n_{u,m}-1), u_1(k-1), \ldots u_m(k-1), 1] \end{aligned} \tag{3.19}$$

$$\theta_i = [\mu_{\ell i,1}, \ldots, \mu_{\ell i,n_{y,1}}, \mu_{\ell i,n_{y,1}} + 1, \ldots, \mu_{\ell i, \sum_{j=1}^{p}(n_{y,j+1})}, \ldots, \mu_{\ell i, \sum_{j=1}^{p}(n_{y,j+1}) + \sum_{j=1}^{m} n_{u,j}}, \eta_{\ell i,1}, \ldots, \eta_{\ell i,m}, \varphi_{\ell i}]^T \tag{3.20}$$

Let

$$\psi(k-1) = \left[\bar{\beta}_{\ell 1}\psi(k-1), \bar{\beta}_{l2}\psi(k-1), ..., \bar{\beta}_{\ell c}\psi(k-1)\right]^T \tag{3.21}$$

$$\Theta^{\ell} = \left[\theta_1^{\ell}, \theta_2^{\ell}, ..., \theta_c^{\ell}\right]^T \tag{3.22}$$

Then eq(3.18) can be reexpressed as

$$y_{\ell}(k) = \psi_{\ell}(k-1)^T \Theta \tag{3.23}$$

In this context, $\Psi(k-1)$ represents the regression vector, and $\Theta$ stands for the unknown resultant parameter vector, which can be determined using the Weighted Least Square (WLS) algorithm, as outlined below:

$$\psi_l = \begin{bmatrix} \psi_{l1}^T \\ \psi_{l2}^T \\ \vdots \\ \psi_{lN}^T \end{bmatrix}, y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}, W_i = \begin{bmatrix} \mu_{i1} & 0 & \cdots & 0 \\ 0 & \mu_{i2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mu_{iN} \end{bmatrix}$$

The parameters $a_i$ and $b_i$ are combined into a single parameter vector denoted as $\theta_i^{\ell}$. This vector encompasses the consequent parameters specific to the $i^{th}$ cluster.

$$\theta_i^{\ell} = [a_i \; b_i]^T \tag{3.24}$$

Assuming that each distinct local linear model corresponds to a cluster, the consequent parameter vector is represented as $\theta_i$, where $i = 1, 2, ..., p$. The membership degrees $\mu_{ik}$ from the fuzzy partition are employed to determine the importance of the data pair $(x_k, y_k)$.

In case the columns of the extended regressors $X_e$ are linearly independent and $\mu_{ik} > 0$ holds true for $1 \leq k \leq N$, then

$$\Theta_i^{\ell} = \left[\psi_{\ell e}^T W_i \psi_{\ell e}\right]^{-1} \psi_{\ell e}^T W_i y \tag{3.25}$$

$\Theta_i$ is the WRLS solution of $y = \psi_e \Theta + \varepsilon$, where the parameter $a_i$ and $b_i$ are expressed as:

$$a_i = [\theta_1, \theta_2, ..., \theta_p], b_i = \theta_{p+1} \tag{3.26}$$

## 3.7 State Space Model

T-S models are linearized to be a state space model, Eq(3.18) is re-expressed as [44]

$$y_\ell(k) = \hat{\mu}_\ell \psi_\ell(k) + \hat{\eta}_\ell u(k) + \hat{\varphi}_\ell, \ell = 1,2,..,p \tag{3.27}$$

with

$\hat{\mu}_\ell(k) = \sum_{i=1}^{c} \bar{\beta}_{\ell i}(\psi_\ell, u)\, \mu_{\ell i}$

$\hat{\eta}_\ell(k) = \sum_{i=1}^{c} \bar{\beta}_{\ell i}(\psi_\ell, u)\, \eta_{\ell i}$

$\hat{\varphi}_\ell(k) = \sum_{i=1}^{c} \bar{\beta}_{\ell i}(\psi_\ell, u)\, \varphi_{\ell i}$

Define

$$\bar{X} = [y_1(k), \ldots, y_1(k-n_{y,1}), \ldots y_p(k), \ldots,$$
$$y_p(k-n_{y,p}), u_1(k-1), \ldots, u_1(k-n_{u,1}), \ldots, u_m(k-n_{u,m})]^T$$

Subsequently, T-S fuzzy models for MIMO nonlinear processes are converted into a linear time-varying state space model

$$\begin{cases} \bar{X}(k) = A(k)\bar{X}(k) + B(k)u(k) \\ \bar{y}(k) = C(k)\bar{X}(k) \end{cases} \tag{3.28}$$

In this context, $A(k)$ stands for the state matrix, $B(k)$ denotes the control matrix, and $C(k)$ represents the output matrix. With these definitions, it becomes apparent that

$$A = \begin{bmatrix} \hat{\mu}_{1,1} & \hat{\mu}_{1,2} & \cdots & \cdots & \cdots & \cdots & \hat{\mu}_{1,s-1} & \hat{\mu}_{1,s} \\ 1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 & 0 \\ 0 & 1 & \cdots & \cdots & \cdots & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \hat{\mu}_{p,1} & \hat{\mu}_{p,2} & \cdots & \cdots & \cdots & \cdots & \hat{\mu}_{p,s-1} & \hat{\mu}_{p,s} \\ 0 & 0 & \cdots & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}_{s*s} \tag{3.29}$$

$$B = \begin{bmatrix} \hat{\eta}_{1,1} & \hat{\eta}_{1,2} & \dots & \dots & \hat{\eta}_{1,q} \\ 0 & 0 & \dots & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \hat{\eta}_{p,1} & \hat{\eta}_{p,1} & \dots & \dots & \hat{\eta}_{p,q} \\ 0 & 0 & \dots & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & \dots & \dots & 0 \\ 0 & 0 & \dots & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}_{s*q} \tag{3.30}$$

$$C = \begin{bmatrix} 1 & 0 & \dots & \dots & \dots & \dots & \dots & \dots & 0 \\ 0 & 0 & \dots & 1 & 0 & \dots & \dots & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & \dots & \dots & 1 & 0 & \dots & 0 \end{bmatrix}_{p*s} \tag{3.31}$$

## 3.8 Summary

Fuzzy clustering stands as a potent unsupervised technique for both data analysis and model construction. Within this chapter, we have provided a comprehensive survey of the commonly employed fuzzy clustering algorithms. We've illustrated that by integrating the foundational c-means iterative approach with adaptive distance metrics, the identification of clusters with diverse geometries becomes attainable. Furthermore, we've delved into the discourse surrounding the selection of pivotal user-defined parameters, including the determination of the number of clusters and the tuning of the fuzziness parameter.

# Chapter 4

# Model based Predictive Control

## 4.1 Introduction

Model Predictive Control (MPC) is a control strategy grounded in a dynamic model of the plant process to formulate the controller's design. It is alternatively recognized as Receding Horizon Control (RHC) or Moving Horizon Optimal Control.

Within MPC, the control input is computed by solving an online optimization problem spanning a finite prediction horizon at every control step. This prediction horizon signifies a time-window of future steps wherein control inputs are optimized. The optimization objective is to determine an optimal control sequence that minimizes a designated cost function while respecting constraints on system dynamics, control inputs, and states.

The outcome of this optimization process furnishes the optimal control sequence for the entire prediction horizon, yet only the initial control input is enacted on the plant during the present time step. This approach is often termed "shifting" or "receding horizon," denoting the forward motion of the prediction horizon as control steps unfold.

Subsequently, with each ensuing control step, the optimization problem is recomputed using the updated plant state as the starting point. This adaptability empowers the controller to accommodate changes in system dynamics and external disturbances, enabling informed control decisions based on real-time information.

MPC presents several benefits, encompassing the capacity to manage constraints on system variables, the consideration of forthcoming system behavior for enhanced decision-making, and the flexibility to incorporate intricate models and performance criteria. It finds broad application across diverse industries such as process control, automotive control, power systems, and robotics, where predictive capabilities and optimization are pivotal for efficacious control strategies [49].

The MPC technology that originated from the late 1960*s* has been developed to a quite mature level. Many results on stability, optimality and performance tuning are presented

in the literature. However, this technology is challenged by the uncertainty existing in the system, such as model structure error, state estimation error, actuator error, measurement error and exogenous disturbances. Stability, performance and constraint satisfaction should be checked under the uncertainty scenarios[29]. Robustness of the controller is the ability to maintain these properties when there are discrepancies in the information on which the controller design relies, e.g. model mismatch and actuator distortion.

Satisfying constraints holds significant importance in numerous process plants, driven by considerations of safety, productivity, and environmental protection.

Within the context of quality control in DWDS, it is imperative to uphold chlorine residuals within specified upper and lower limits. As elaborated in chapter , the system is influenced by various uncertain factors that can impact the attainment of these set objectives.

The controller outputs derived through computation, often reliant on a plant model, might fail to adhere to plant output constraints due to disparities between the model and the actual plant behavior. Control inputs that are feasible according to computations might become unfeasible when executed on the plant if the controller lacks robustness. The primary aim of designing a robust model predictive controller in this chapter is to ensure the robustness of maintaining output constraints in the face of uncertainties within the system, encompassing factors like uncertainties in models, actuators, and measurements. Without prior declaration, robust MPC refers to MPC with output constraint satisfaction under uncertainties in this thesis.

Predictive control design finds extensive adoption in the petrochemical and related industries, particularly where the management of diverse constraints is imperative. This approach is essential because operational efficiency often entails operating at or near the boundaries of the permissible states and controls[9].

It comprises four fundamental components: (i) a process model that characterizes the system, (ii) an objective function delineating the goal through predictions of future inputs and outputs, often taking the form of a quadratic equation, (iii) optional constraints imposed on both system and control variables, and (iv) an optimization approach that constitutes a convex quadratic problem [21].

## 4.2 Formulation of Model Predictive Controller

The concept of MPC is depicted in Figure4.1. The current time instant is $t$ and the present output is $y(t)$. The previous output is also shown in the figure. A finite time horizon $H_p$, output prediction horizon, is defined and the system outputs over the finite horizon $[t+1, t+H_p]$ are to be controlled. $s(t)$ is the set-point trajectory that may be varying depending on the

operation of the plant.

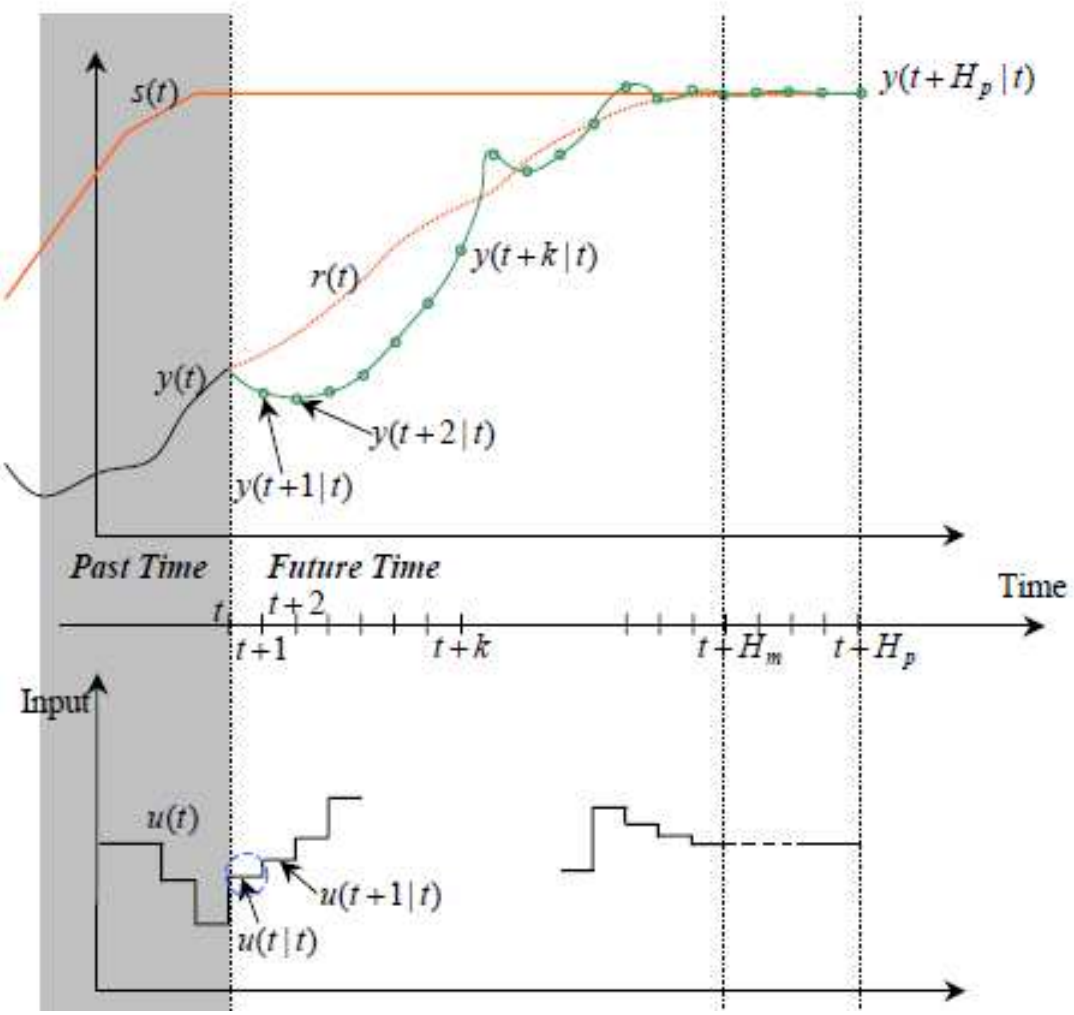


Fig. 4.1 Operation of the MPC

The objective of the controller is to track the set-point trajectory (the constant set-point regulation problem is also included as a special case). A reference trajectory $r(t)$ is defined starting from the current time instant and extending over the prediction horizon $H_p$. It considers an ideal or desired tracking trajectory from the current output to the set-point trajectory.

Actually, the reference trajectory $r(t)$ defines a closed-loop behaviour of the controlled plant, which is the performance of the controller. $r(t)$ can be calculated before running MPC by specifying the performance of tracking set-point trajectory $s(t)$ from the current output $y(t)$. For example, it is very convenient to define a first-order or second-order system to describe the tracking performance and to take the time constant and damping ratio of the reference

system as the tuning knobs of the MPC performance[34].
The vector of outputs over prediction horizon is defined as:

$$\hat{y} = \left[y(t+1|t) \cdots y(t+H_p|t)\right]^T \tag{4.1}$$

where $y(t+i|t), = 1...H_p$ stands for model output at $t+i$, and such model prediction is performed at time instant $t$.

Control input vector over the prediction horizon is defined as:

$$\hat{U} = [u(t|t) \cdots u(t+H_m-1|t) \;\; u(t+H_m-1|t) \cdots u(t+H_m-1|t)]^T \tag{4.2}$$

where $u(t+i|t), = 1...H_m$ are the controller output series produced by the controller at time instant $t$, $H_m$ is the input horizon and the control input keeps the same value from the end of the input horizon to the end of the prediction horizon. There may be output and input constraints on the control input and output values.
Let the following inequalities denote the output and input constraints, respectively:

$$C_y\left(\hat{Y}\right) \leq 0 \tag{4.3}$$

$$C_u\left(\hat{U}\right) \leq 0 \tag{4.4}$$

By using the plant model explicitly the output over the prediction horizon $\hat{Y}$ can be obtained knowing the current and past states $x(t)$ , past inputs $u(t)$:

$$\hat{Y} = M\left(x(t), u(t), \hat{U}, \theta\right) \tag{4.5}$$

where $M(.)$ denotes the model of the plant, which could be linear or non-linear, time-invariant or time-varying, and $\theta$ is the model parameters. In order to track the reference trajectory, a performance index $V(.)$ is defined, which is also referred to as cost function because of the inclusion of the control cost in the function. Usually $V(.)$ can be defined by using $L_1$, $L_2$ or $L_\infty$ norms to measure the distance between the predicted outputs and reference values. If the model is linear and $L_2$ norm is applied, a final optimization problem is the quadratic one:

$$\hat{U} = \arg\min V(\hat{U}) = (\hat{Y} - R)^T P(\hat{Y} - R) + \hat{U}^T Q\hat{U} \tag{4.6}$$

subject to :

$$C_y\left(\hat{Y}\right) \leq 0$$

$$C_u\left(\hat{U}\right) \leq 0$$

where $R = [r(t+1)...r(t+H_p]^T$ are the references values, $C_y(.)$ and $C_u(.)$ stand for the constraints on the outputs and inputs, respectively.
In order to guarantee stability extra constraints may be added as well. For instance, constraint on terminal output over the prediction horizon.
The MPC algorithm can now be stated as:

**Algorithm of MPC**

1. At time $t$ , obtaining the current state of the plant and other information in the past;
2. Solving optimization problem described in(4.6) ;
3. Applying only the first control value in the control sequence, that is
   $u(t) = u(t\,|t\,)$ ;
4. $t \leftarrow t+1$. Go to step 1.

There are also other formulations of MPC such as for unstable system, continuous system and non-quadratic performance index function[34]. There are still some applications in practice where there is no clearly specified reference for the output.
For instance, in chlorine residual concentration control in the DWDS the objective is to maintain a minimum chlorine residual concentration in the water distribution network and at the same time keep the concentration under the upper limit. The particular output trajectory is not very important in this case. In other words, the reference is a zone instead of a trajectory, which is shown in figure 4.2

Two methods can be used to formulate the zone-referenced MPC problem. First, there is no penalty for outputs in the performance index and the reference zone is to be implemented as hard constraints:

$$Y^{\min} \leq \hat{Y} \leq Y^{\max} \tag{4.7}$$

where $Y^{min}$ and $Y^{max}$ define the upper and lower limits of the reference zone. Constraint described by(4.7) can be added into optimization problem described in(4.6) explicitly. This is called the hard-constraint method.
The second method is so-called soft-constraint method that is to only penalize value of the outputs out of the zone by the following function:

$$\mathrm{V_z}(\hat{Y}) = \sum_{i=1}^{H_p} [|y(t+i\,|t) - r_l(t+i)\,| + |y(t+i\,|t) - r_u(t+i)\,|] \tag{4.8}$$

where $r_l(t+k)$,$r_u(t+k)$ are are the lower and upper value of the zone output reference, respectively, for $k = 1,...,H_p$

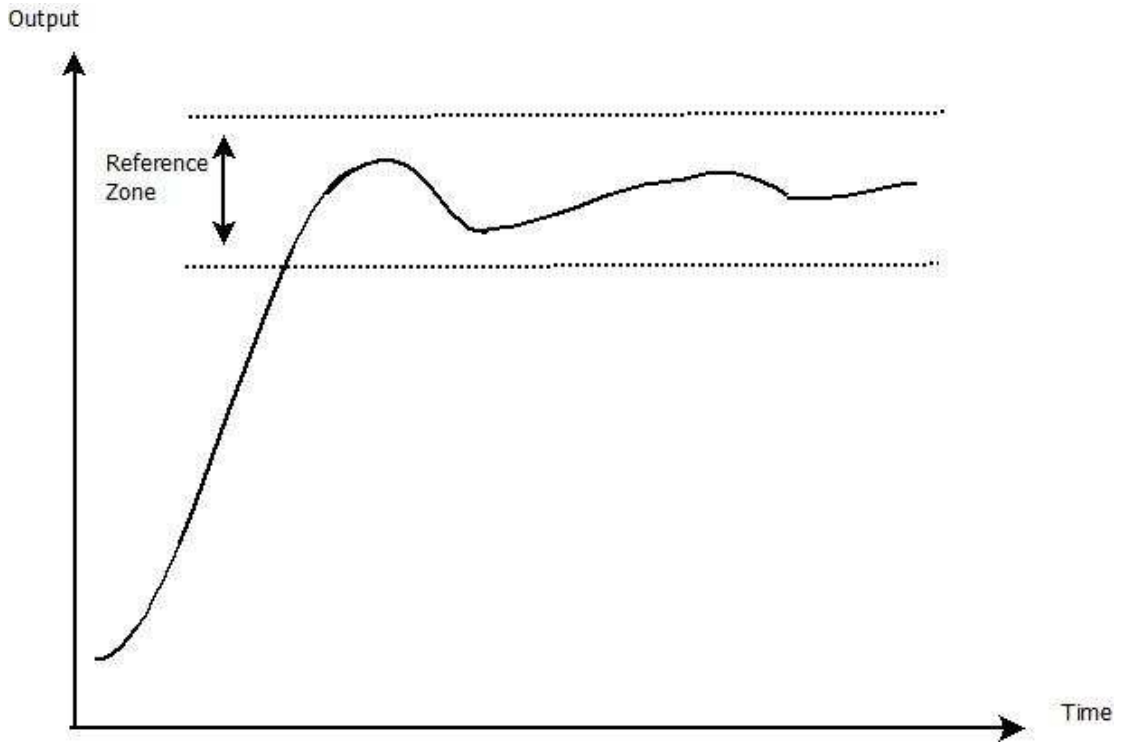


Fig. 4.2 Zone Reference MPC

Considering the predicted output $y(t+i|t)$ at time instant $t+i$, if its value is greater than the upper limits then:

$$\begin{aligned} &|y(t+i|t)-r_l(t+i)|+|y(t+i|t)-r_u(t+i)|= \\ &y(t+i|t)-r_l(t+i)+y(t+i|t)-r_u(t+i)= \\ &2*[y(t+i|t)-r_u(t+i)]+r_u(t+i)-r_l(t+i) \end{aligned} \tag{4.9}$$

and if $y(t+i|t)$ is less than lower limit of the zone then:

$$\begin{aligned} &|y(t+i|t)-r_l(t+i)|+|y(t+i|t)-r_u(t+i)|= \\ &-y(t+i|t)+r_l(t+i)-y(t+i|t)+r_u(t+i)= \\ &2*[r_l(t+i)-y(t+i|t)]+r_u(t+i)-r_l(t+i) \end{aligned} \tag{4.10}$$

if $y(t+i|t)$ is within the zone then:

$$\begin{aligned} &|y(t+i|t)-r_l(t+i)|+|y(t+i|t)-r_u(t+i)|= \\ &y(t+i|t)-r_l(t+i)-y(t+i|t)+r_u(t+i)= \\ &r_u(t+i)-r_l(t+i) \end{aligned} \tag{4.11}$$

Hence, when the optimal control is achieved the performance index takes the minimum value:

$$V_{z\min} = \sum_{i=1}^{H_p} [r_u(t+i) - r_l(t+i)] \tag{4.12}$$

The method is a kind of soft constraint method because the output violations are penalised instead of fulfilling hard output constraints. The optimization of absolute value performance index can be converted into linear programming. This kind of formulated controller can be used as the remedy controller if the first method does not work when there is no feasible solution in the optimal problem, which will be presented in detail in the last section of this chapter.
Equation (4.8) gives one type of penalty function that can be conveniently used in MPC formulation.

## 4.3 Takagi-Sugeno Fuzzy Model Predictive Control

The exploration of local linearization within Takagi-Sugeno fuzzy models is pursued to broaden the scope of applicability for linear model predictive controllers[9].

The optimization challenge faced by MPC at time step $k$ can be formulated as follows:

$$J = \sum_{i=1}^{H_p} \|r(k+i) - \hat{y}(k+i)\|_P^2 + \sum_{i=1}^{H_c} \|u(k+i-1)\|_Q^2 \tag{4.13}$$

In this context $P$ and $Q$ represent positive definite weight matrices.

Within the realm of linear model-based predictive control, a linear model is employed to forecast the output $\hat{y}$ based on the sequence of control signals $\hat{u}(k,\cdots,k+H_p)$, where $H_p$ signifies the prediction horizon.

A linear model in state-space description is given by:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) \end{cases} \tag{4.14}$$

and by referring to the objective function defined in Equation (4.13), the constrained linear model-based predictive control can be obtained by resolving the quadratic problem:

$$\min_{\Delta\tilde{u}} \left\{ \frac{1}{2} \Delta\tilde{u}^T H \Delta\tilde{u} + c^T \Delta\tilde{u} \right\} \tag{4.15}$$

with

$$\begin{cases} H = 2\left(R_u^T P R_u + Q\right) \\ c = 2\left[R_u^T P^T \left(R_x A x(k) - \tilde{r}\right)\right]^T \end{cases} \tag{4.16}$$

and verifying the constraints on $u$, $\Delta u$, and $y$:

$$\Lambda \Delta \tilde{u} \leq \omega \tag{4.17}$$

with

$$\Lambda = \begin{bmatrix} I_{\Delta u} \\ -I_{\Delta u} \\ I^{H_p m} \\ -I^{H_p m} \\ R_u \\ -R_u \end{bmatrix}, \quad \omega = \begin{bmatrix} u^{\max} - I_u u(k-1) \\ -u^{\min} - I_u u(k-1) \\ \Delta u^{\max} \\ -\Delta u^{\min} \\ y^{\max} - R_x A x(k) \\ y^{\min} - R_x A x(k) \end{bmatrix} \tag{4.18}$$

$I^{H_p m}$ is a $(H_p m \times H_p m)$ unity matrix. The matrices $R_x$, $R_u$, $I_u$ and $I_{\Delta u}$ are defined:

$$R_x = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{H_p-1} \end{bmatrix} \tag{4.19}$$

$$R_u = \begin{bmatrix} CB & 0 & \cdots & 0 \\ CAB & CB & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{H_p-1}B & CA^{H_p-2}B & \cdots & CA^{H_p-H_c}B \end{bmatrix} \tag{4.20}$$

## 4.4 Summary

Due to complexity of the problem, e.g. non-linearity, constraints and uncertainty, it is difficult to perform a stability analysis of the designed controller. However, it is convincing that controller stability can be verified by simulation studies provided sufficiently accurate DWDS numerical models.
This part showcases the practical implementation of constrained multivariable fuzzy predictive control to maintain the chlorine concentration within drinking water networks. The computation of control signals aims to optimize the forthcoming behavior of the underlying processes, emphasizing precise set-point accuracy.

The creation of the Takagi-Sugeno fuzzy system comprises two pivotal phases. Initially, the structure is identified, followed by the parameter adaptation. The learning algorithm integrates a self-adjusting mechanism for learning rates, which is contingent on the evolution of the cost function. The outcomes of simulations underscore the efficacy of the proposed fuzzy control strategy in proficiently regulating the chlorine concentration within the Drinking Water Distribution System (DWDS).

# Chapter 5

# Data-driven based DWDS Monitoring

## 5.1 Introduction

Identifying novel events or anomalies presents a formidable challenge in the realm of signal classification. There exists no universally optimal model for novelty detection, and the efficacy of distinct models is contingent on multiple factors, encompassing the chosen methodology and the statistical characteristics of the data.

Multivariate Statistical Process Control (MSPC) is a technique commonly used for performance monitoring in bioprocesses and industrial processes. However, when traditional MSPC approaches are applied to industrial processes, they may not always be as effective as expected. This can be due to the unfitness of the process statistical model.

Industrial processes can be complex and dynamic, making it difficult to accurately capture their statistical properties using traditional models. Factors such as non-linear behavior, time-varying dynamics, and complex interactions between process variables can lead to deviations from the expected statistical patterns.

To address this limitation, various approaches have been developed to improve the effectiveness of novelty detection in industrial processes. These include:

-Nonlinear MSPC: Introducing non-linear models or considering non-linear relationships between variables can capture more complex behaviors and improve the detection of anomalies.

-Data-driven methods: Instead of relying solely on pre-defined statistical models, data-driven approaches use historical data to learn the normal patterns of the process and identify deviations from these patterns. Machine learning techniques, such as clustering, classification, or anomaly detection algorithms, can be employed for this purpose.

-Hybrid approaches: Combining multiple models or approaches, such as integrating traditional statistical models with machine learning algorithms, can enhance the performance of novelty detection by leveraging the strengths of different methods.

-Domain knowledge: Incorporating domain knowledge and expert insights into the modeling and detection process can help in defining relevant features, identifying critical process variables, and improving the interpretability and effectiveness of the novelty detection.

It is important to carefully analyze the specific characteristics of the industrial process and select or develop an appropriate novelty detection approach that can accommodate the unique statistical properties and dynamics of the system. Continuous monitoring and evaluation of the detection performance are also crucial to ensure the effectiveness of the chosen method in detecting novel events and anomalies.

In numerous scenarios, the process signals under observation are heavily laden with measurement or operational noise, making it challenging to extract authentic signals from the noise-contaminated process data using conventional latent projection techniques like Principal Component Analysis (PCA) or Partial Least Squares (PLS)[30].

## 5.2 Linear Principal Component Analysis

Principal Component Analysis (PCA) is a commonly employed technique for transforming features to diminish the dimensionality of a dataset. The primary objective of PCA is to identify basis vectors within a subspace that optimally minimizes the least square reconstruction error. Let $\mathrm{x} = (\mathrm{x}_1, \cdots, \mathrm{x}_N)$ be a $n-$dimenstional data matrix of points. PCA facilitates the projection of correlated high-dimensional data onto a hyperplane by utilizing solely the initial few $q$ non-zero eigenvalues and their corresponding eigenvectors derived from the covariance matrix $F$, $F = U\Gamma U^T$ where $\Gamma$ incorporates the eigenvalues $\lambda_i$ of $F$ along its diagonal, arranged in descending order, while $U$ denotes the matrix comprising the corresponding eigenvectors. The vector $y = W^T(\mathrm{x})$ is a $q-$dimentional reduced representation of the observed vector $\mathrm{x}_i$ where the $W$ weight matrix contains $q$ principal orthonormal axes in its column $W = U_q \Lambda_q^{1/2}$

In probabilistic PCA, the underlying assumption is that the data originates from a probability model. As a result, the probabilistic PCA model can be expanded to encompass a mixture model, wherein the data is conceived as emerging from multiple components mixed in diverse proportions. Each of these components can be represented by a Gaussian probability density function:

$$p(x) = \frac{1}{(2\pi)^{n/2} |A|^{1/2}} \exp\left( -\frac{1}{2} (x - \bar{x})^T A^{-1} (x - \bar{x}) \right),$$

where $A = \delta^2 I + WW^T$ represents the adapted covariance matrix with $I$ denoting the identity matrix. The matrix $W$ is determined in the same manner as in the conventional PCA

algorithm, while $\delta^2$ is computed as the mean of the eigenvalues that have been omitted.

$$\delta^2 = \frac{1}{n-q} \sum_{i=q+1}^{n} \lambda_i$$

## 5.3 Nonlinear Principal Component Analysis

In recent times, there has been a notable surge in interest surrounding Nonlinear Principal Component Analysis (NLPCA), which represents an evolution beyond the traditional linear PCA. Numerous scholars have embraced a neural-oriented methodology to construct the NLPCA model, originally pioneered by Kramer[27].

When focusing on a solitary non-linear principal component, the arrangement of this network is illustrated in Figure 5.1. To actualize Nonlinear Principal Component Analysis (NLPCA), the auto-associative network integrates three layers positioned between the input and output variables. A transfer function $\Xi 1$ executes the projection of the input column vector, possessing a dimension of $m$, onto the first hidden layer, often referred to as the coding layer. This is represented by the column vector $\left(h_j^{(x)} \mid j = 1, \cdots, r\right)$ of dimension $r$ ( $r$ is the number of neurons in the first hidden layer):

$$h_j^{(x)} = f_1 \left( \sum_{i=1}^{m} v_{ij}^{(x)} x_i + b_j^{(x)} \right) \tag{5.1}$$

$v^{(x)} = \left[ v_1^{(x)} \cdots v_m^{(x)} \right]$, with $v_i^{(x)}, i = 1, \cdots, m$ the vector comprises the $r$ bias parameters. The subsequent transfer function $f_2$ carries out the projection of the output data from the initial hidden layer (coding layer) to the *bottleneck layer* which encompasses a singular neuron signifying the nonlinear principal component $t$. The transfer function $f_1$ typically embodies a nonlinear nature (frequently the hyperbolic tangent function or sigmoid function), whereas the function $f_2$ commonly takes the form of the identity function. $f_2(x) = x$ [6]

$$t = f_2 \left( \sum_{j=1}^{r} w_j^{(x)} h_j^{(x)} + \overline{b}^{(x)} \right) \tag{5.2}$$

Subsequently, the nonlinear transfer function $f_3$ employed to project the data from $t$ to the final hidden layer, known as the decoding layer. This operation yields a set of values $\left(h_j^{(t)} \mid j = 1, \cdots, r\right)$ where $r$ corresponds to the number of neurons in the third hidden layer:

$$h_j^{(t)} = f_3 \left( w_j^{(t)} t + b_j^{(t)} \right) \tag{5.3}$$

The last transfer function $f_4$ is the identity function which projects the outputs data from $h_j^{(t)}$ to $\hat{x}_i$: the output column vector of dimension $m$:

$$\hat{x}_i = f_4\left(\sum_{j=1}^{r} v_{ij}^{(t)} h_j^{(t)} + b_i^{(t)}\right) \tag{5.4}$$

The cost function $E = \frac{1}{N}\sum_{k=1}^{N}(x(k)-\hat{x}(k))^2$ is minimized to find the optimal values of the $v_i^{(x)}$ , $b^{(x)}$, $w_i^{(x)}$, $\overline{b}^{(x)}$, $w^{(t)}$ , $b^{(t)}$, $v^{(t)}$, and $b_i^{(t)}$

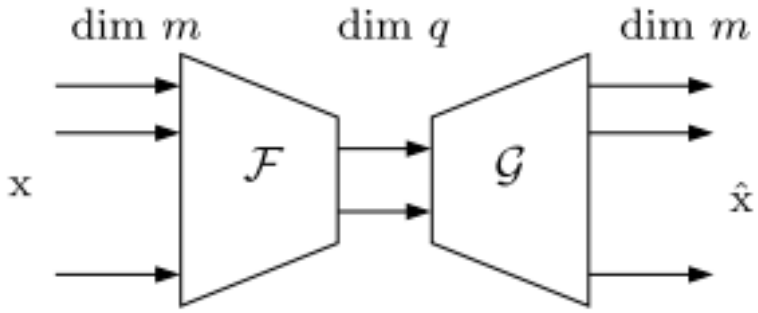


Fig. 5.1 Bottleneck Neural Network

## 5.4 The ConceptT Of Bottleneck Neural Network and Nonlinear Variable Reconstruction

A Bottleneck Neural Network (BNN) is employed for the purpose of identifying and mitigating correlations between variables, thereby aiding in dimensionality reduction, data visualization, and exploratory analysis. BNN is capable of revealing both linear and nonlinear correlations without being constrained by the specific nature of nonlinearities within the data. The neural network configuration encompasses five layers: the input layer, mapping layer, bottleneck layer, de-mapping layer, and output layer. This class of neural networks is distinct in that it can acquire the principal components without necessitating the direct computation of eigenvalues and eigenvectors from the sample covariance matrix.[8].
In order to handle the difficulties associated with the training task of the full bottleneck neural network, especially for the case of complex processes, we propose to train two subnetworks, namely the *projective subnetwork* and the *reconstruction network* figure.5.2. This idea is proposed for the first time by[15] where two 3-layer feedforward neural networks are used in conjunction with the principle curves algorithm to find the projection scores in the nonlinear principal component subspace. In the same way,[50] proposed to used radial ba-

sis function network (RBF) instead of 3-layer feedforward neural networks. In this paper, we propose to use a fuzzy clustering algorithm to find the nonlinear projection scores and two feedforward neural networks (FNN) , one for the projection subnetwork and one for the reconstruction subnetwork.

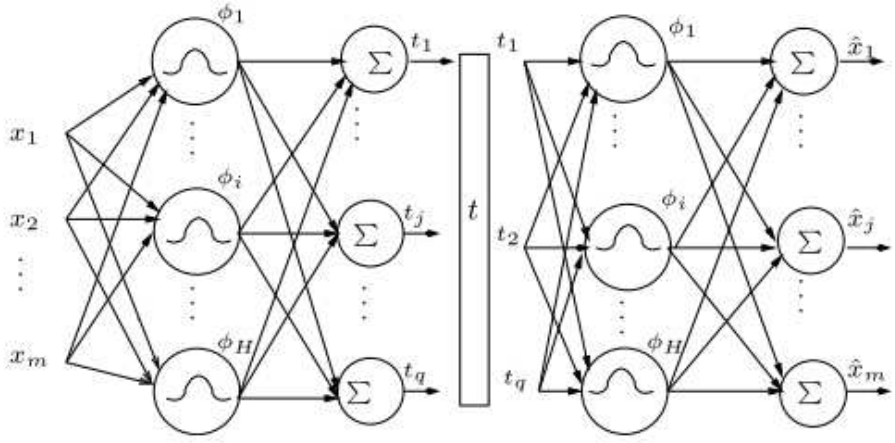


Fig. 5.2 The architecture of the NLPCA based on RBF networks

To identify multiple linear subspaces, referred to as local PCA models, clustering is conducted within low-dimensional subspaces instead of the original high-dimensional space, where clusters may not exhibit clear separation.

The computation of the prior probability for each cluster can be expressed as:

$$\alpha_i = \frac{1}{N}\sum_{k=1}^{N}\gamma_{i,k}$$

where $\gamma$ is degree of membership.
The cluster centers are determined from

$$v_i^x = \frac{\sum_{k=1}^{n}\left(\gamma_{i,k}\right)^m\left(x_k - W_i\left\langle y_{i,k}\right\rangle\right)}{\sum_{k=1}^{n}\left(\gamma_{i,k}\right)^m}$$

where the expectation of the *latent variables* is

$$\left\langle y_{i,k}\right\rangle = M_i^{-1}W_i^T\left(x_k - v_i^x\right)$$

the $q \times q$ matrix $M_i = \sigma_{i,x}^2 I + W_i^T W_i$ and the fuzzy weighting coefficient $m = 2$. The new value of $W_i$ can be computed by:

$$\tilde{W}_i = F_i W_i\left(\sigma_{i,x}^2 I + M_i^{-1}W_i^T F_i W_i\right)^{-1}$$

The covariance matrix $F_i$ can be computed by

$$F_i = \frac{\sum_{k=1}^{N} \left(\gamma_{i,k}\right)^m \left(x_k - v_i^x\right)\left(x_k - v_i^x\right)^T}{\sum_{k=1}^{N} \left(\gamma_{i,k}\right)^m}$$

the new value of $\sigma_{i,x}^2$ is

$$\sigma_{i,x}^2 = \frac{1}{q} tr\left(F_i - F_i \tilde{W}_i M_i^{-1} \tilde{W}_i^T\right)$$

the fuzzy covariance matrix is

$$A_i = \sigma_{i,x}^2 I +_i \tilde{W}_i^T$$

The fuzzy clustering is an iterative algorithm that minimizes the objective function:

$$J = \sum_{i=1}^{c} \sum_{k=1}^{N} \left(\gamma_{i,k}\right)^m D_{i,k}^2 + \sum_{k=1}^{N} \lambda_i \left(\sum_{i=1}^{c} \gamma_{i,k} - 1\right)$$

where the $D_{i,k}$ is a distance measure comprises three components: the prior probability of the cluster, the distance between the k$th$ data point and the centroid $v_i$ of cluster $i$, and the distance between the cluster prototype and the data within the subspace.

### 5.4.1 Fault detection

The process of sensor fault detection using BNN involves the evaluation of residuals, as depicted in Figure 5.3. The squared Prediction Error (SPE) is a statistical measure utilized to assess the adequacy of the BNN model's fit. At time k, the detection index SPE can be expressed as:

$$SPE(k) = e^T(k)e(k) = \sum_{i=1}^{m} e_i^2(k) \tag{5.5}$$

The SPE statistic distribution can be accurately approximated by

$$SPE \propto g\mathscr{X}_h^2 \tag{5.6}$$

The weight $g$ and the degree of freedom $h$ can be estimated through the matching of moments of the mean ($m$) and variance ($v$) of the cumulants :

$$g = \frac{v}{2m} \tag{5.7}$$

and

$$h = \frac{2m^2}{v} \tag{5.8}$$

Therefore, the upper control limit can be determined as a result of this calculation:

$$\delta_\alpha = \frac{v}{2m}\mathscr{X}^2_{1-\alpha}\left(\frac{2m^2}{v}\right) \tag{5.9}$$

when the predefined level of significance $\alpha$ is applied, an abnormal situation is identified in the following scenario:

$$SPE(k) > \delta^2_\alpha \tag{5.10}$$

when $\delta^2_\alpha$ represents a confidence limit for the estimated SPE (Squared Prediction Error) using historical data, to enhance detection capabilities, the exponentially weighted moving average (EWMA) filter can be employed. This application of the EWMA filter helps mitigate the frequency of false alarms arising from noise. The typical mathematical expression for the EWMA filter in relation to residuals is:

$$\overline{\mathbf{e}}(k) = (I-\beta)\,\overline{\mathbf{e}}(k-1) + \beta\mathbf{e}(k) \tag{5.11}$$

$$\overline{SPE}(k) = \|\overline{\mathbf{e}}(k)\|^2 \tag{5.12}$$

Here, $\beta = \gamma I$ represents a diagonal matrix with diagonal elements $\gamma$, which serve as forgetting factors for the residuals. The control limit is derived based on reference data. If the Squared Prediction Error (SPE) surpasses its control limit, it indicates that the system is in a faulty state. This can be attributed to a shift in the correlation structure of the process variables, thereby explaining the occurrence.

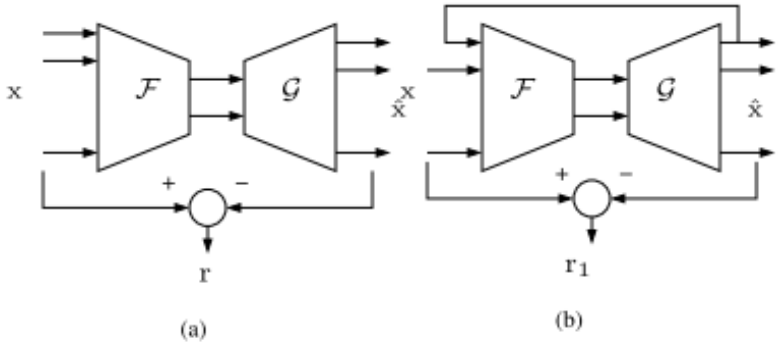


Fig. 5.3 Bottleneck Neural Network for Fault Detection

### 5.4.2 Fault identification

Upon detecting a fault, it becomes imperative to pinpoint the specific variable that is contributing to the issue. This process is referred to as fault localization. Numerous approaches for identifying the variable responsible for defects have been suggested in existing literature. In the context of this study, our focus is directed towards the technique known as "reconstruction fault."

#### Contribution plots

In the conventional methodology employed for fault identification using Principal Component Analysis (PCA), the process relies on the computed contributions to the detection index. Consequently, the variable that exhibits the most substantial contribution to the detection indicator is the one held responsible for influencing the detection index *SPE* (as mentioned in [33].

The contribution of the $j^{th}$ variable to the *SPE* statistic is formally defined as follows:

$$C_j^{[SPE]} = (x_j - \hat{x}_j)^2 \tag{5.13}$$

#### Reconstruction fault

The reconstruction principle involves estimating a specific variable, denoted as $x_j(k)$, within the data vector $x(k)$ at a given time. This estimation is carried out using the PCA model and other variables, as outlined in the work by[2].

Within this framework, there exist three distinct approaches for reconstructing a faulty sensor. In this particular study, the chosen method was the iterative reconstruction algorithm. Assuming knowledge of the fault direction, the reconstruction $\hat{x}_j$ of the $j^{th}$ variable is achieved by utilizing a selected number of principal components through an iterative technique. In this process, the value of the faulty sensor is replaced with the predicted value, and this reconstruction is governed by the equation:

$$\hat{x}_j^{new} = \left[c_{-j}^T 0 c_{+j}^T\right] x + c_{jj} \hat{x}_j^{old} \tag{5.14}$$

where $\hat{x}_j^{old}$ can be considered as a projection of x on the PCs, it can be calculated as by : $\hat{x} = Cx$, where $C = PP^T$ and $\left[c_{-j}^T 0 c_{+j}^T\right]$ is a vector of matrix C which the $j^{th}$ coloumn of $c_{jj}$ is replaced by 0. The iterative process converges to the following formula :

$$x_j^* = \frac{\left[c_{-j}^T 0 c_{+j}^T\right]}{1 - c_{jj}} x \tag{5.15}$$

where $c_{jj} \neq 1$. In the case of $c_{jj} = 1$, the $j^{th}$ cannot be reconstructed by this method.

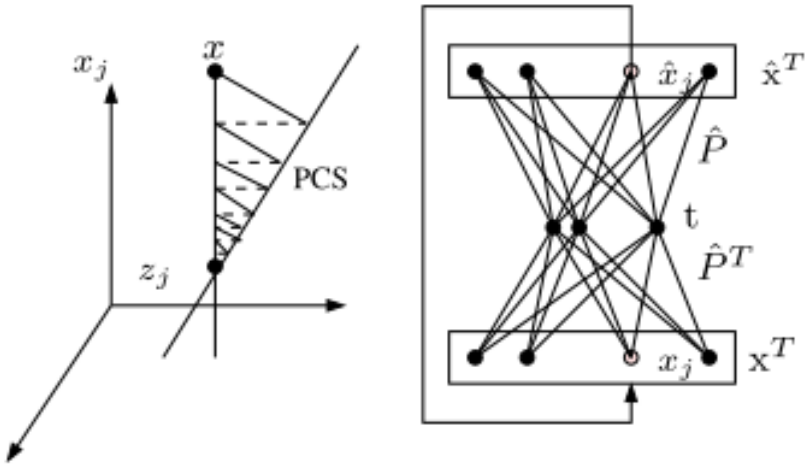


Fig. 5.4 Variable reconstruction in the linear case

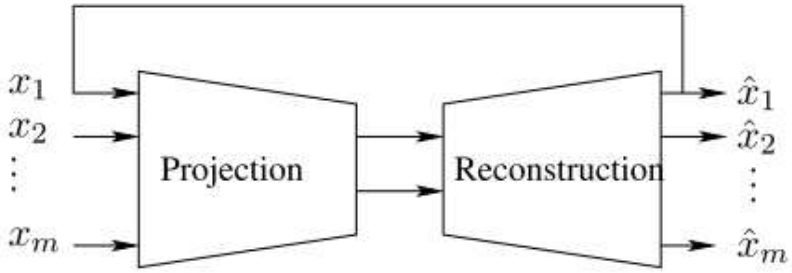


Fig. 5.5 Nonlinear Reconstruction

When dealing with a faulty sensor, a notable decrease in the value of the *SPE* (Squared Prediction Error) is observed both prior to and after the reconstruction process. Nevertheless, in certain scenarios, the act of minimizing the *SPE* can lead to an impact on all data entries. This, in turn, can render the faulty sensor indistinguishable or challenging to identify[23].

#### Sensor validity index

The Sensor Validity Index (SVI) serves as an indicator of sensor performance. It is designed to possess a standardized range, regardless of factors such as the number of principal components, measurement variance, noise levels, or the specific type of faults. The SVI is also expected to effectively distinguish between abnormal operational conditions and situations involving sensor faults, as discussed in[23].

The concept of a Sensor Validity Index $(SVI)$ was introduced to assess sensor integrity. Assuming that only a single sensor fault occurs within the system's processes, this index is employed to ascertain the status of each sensor. Its formal definition is as follows:

$$\eta_j^2 = \frac{SPE(x_j^*)}{SPE(x)} \tag{5.16}$$

where $x_i^*$ is a reconstructed vector.
Apparently, $\eta_j^2$ ranges between $[0,1]$, because the $SPE(x_j) \geq SPE(x)$. When $\eta_j^2$ is close to 0, it indicates that the $j^{th}$ sensor is faulty. On the other hand, when $\eta_j^2$ is close to 1, it means that the sensor variations is consistent with others ones.

## 5.5 Summary

This part introduces a detection approach tailored for systems exhibiting nonlinear behavior. In this study, we present a novel contribution: substituting the linear Principal Component Analysis (PCA) model with a Bottleneck Neural Network (BNN) to enhance the adaptability of this diagnostic method to nonlinear systems.

The utilization of BNN serves the purpose of identifying and eliminating correlations among variables, which aids in tasks such as dimensionality reduction, visualization, and exploratory data analysis.

# Chapter 6

# Experimental Results

## 6.1 Sensor Fault Detection and Isolation :Case Study

The BNN methodology has been applied through simulation on a well-established benchmark system known as the quadruple-tank process.

The quadruple-tank laboratory process, as introduced in Johansson's work in [25], encompasses four interconnected water tanks. Additionally, the system comprises two pumps and their associated valves. The interaction is such that water flows from the upper tanks to the lower ones. One of the pumps is employed to fill the upper left tank and the lower right tank. This task involves using a valve with a fixed position to distribute the pump's capacity between the upper and lower tanks. The second pump, on the other hand, is utilized to fill the upper right tank and the lower left tank[8]. See figure6.1.

The control variables in this context are represented by the pump voltages, denoted as $u_1$ and $u_2$. The system's state variables are established based on the water levels within the tanks, denoted as $x_1, x_2, x_3 \, and \, x_4$ respectively. These water levels are measured in centimeters. It's worth noting that the maximum capacity of each tank is set at 20 cm.

The system's behavior is characterized by the following dynamics:

$$\dot{x}_1 = -\frac{a_1}{A}_2\sqrt{2gx_1} + \frac{a_3}{A}_1\sqrt{2gx_3} + \frac{\gamma_1 k_1}{A_1}u_1 \tag{6.1}$$

$$\dot{x}_2 = -\frac{a_2}{A}_2\sqrt{2gx_2} + \frac{a_4}{A}_2 1\sqrt{2gx_4} + \frac{\gamma_2 k_2}{A_2}u_2 \tag{6.2}$$

$$\dot{x}_3 = -\frac{a_3}{A}_3\sqrt{2gx_3} + \frac{(1-\gamma_2)}{A_3}k_2 u_2 \tag{6.3}$$

$$\dot{x}_4 = -\frac{a_4}{A}_4\sqrt{2gx_4} + \frac{(1-\gamma_1)}{A_4}k_1 u_1 \tag{6.4}$$

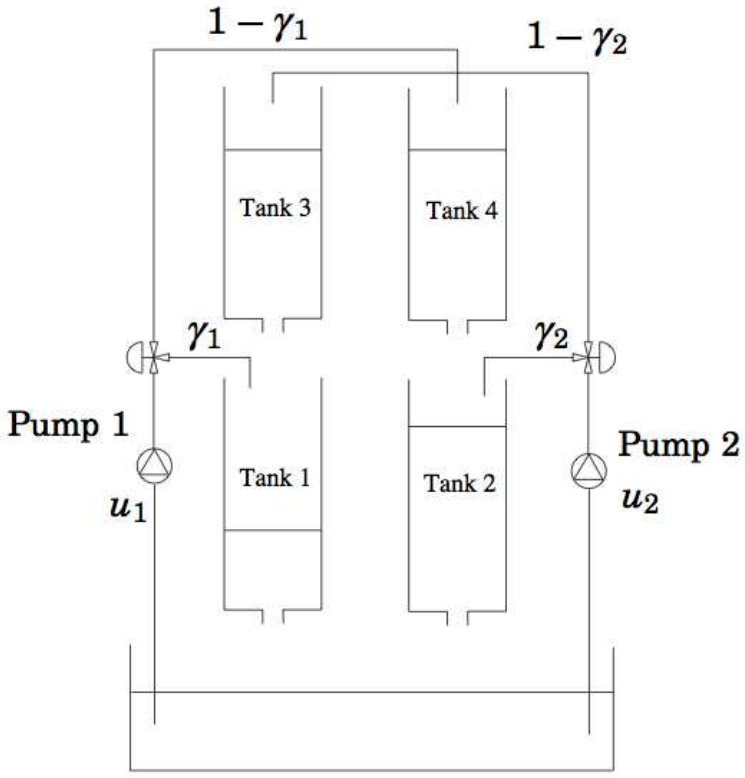


Fig. 6.1 A schematic picture of the quadruple tank process

The dynamics are influenced by the parameters $\gamma_i$, which determine the positions of the valves governing the flow rates into the upper and lower tanks. Additionally, the parameters $A_i$ and $a_i$ denote the cross-sectional area of the tanks and the holes respectively. The control signals, represented as $u_i$, are integral to the system.

The primary objective centers around regulating the water levels in the two lower tanks, denoted as $x_1$ and $x_2$.

The specific numerical values of these parameters can be found in Table6.1. The study of the control of the quadruple-tank process is conducted at a specific operating point, and the corresponding parameter values are presented in Table6.2.

| *Parameters* | *Values* | *Unit* |
|---|---|---|
| $A_1$,$A_2$ | 28 | $cm^2$ |
| $A_3$,$A_4$ | 232 | $cm^2$ |
| $a_1$, $a_2$ | 0.071 | $cm^2$ |
| $a_3$ , $a_4$ | 0.057 | $cm^2$ |
| $k_1$,$k_2$ | 3.33,3.35 | $cm^3/Vs$ |
| $k_c$ | 0.50 | V/cm |
| g | 981 | $cm/s^2$ |

Table 6.1 parameter values of the quadruple-tank

| *Parameters* | *Values* | *Unit* |
|---|---|---|
| $x_1^0$,$x_2^0$ | 12.4,12.7 | cm |
| $x_3^0$,$x_4^0$ | 1.8,1.4 | cm |
| $u_1^0$,$u_2^0$ | 3.00,3.00 | v |
| $\gamma_1$,$\gamma_2$ | 0.70,0.60 | |

Table 6.2 Operating points parameters

Our focus was exclusively directed towards sensor-related faults. The BNN approach we employed is rooted in an auto-associative Neural Network model comprising five layers.

To detect the presence of sensor faults within the quadruple-tank system, we utilized the concept of Squared Prediction Error (SPE). The progression of SPE variation is visually depicted in Figure6.2, showcasing the detection of changes in SPE throughout the simulation.

In Figure6.3 (a), the evolution plot of SPE exhibits two distinct operating regions. Notably, the second region that spans from the $50^{th}$ sample to the simulation's end indicates a sensor fault.

Moreover, the identification of the specific faulty variable is clarified in Figure 6.3 (b). In this figure, the contribution plots for the first variable (which corresponds to the water level

of the first lower tank) demonstrate higher contributions compared to the other variables.

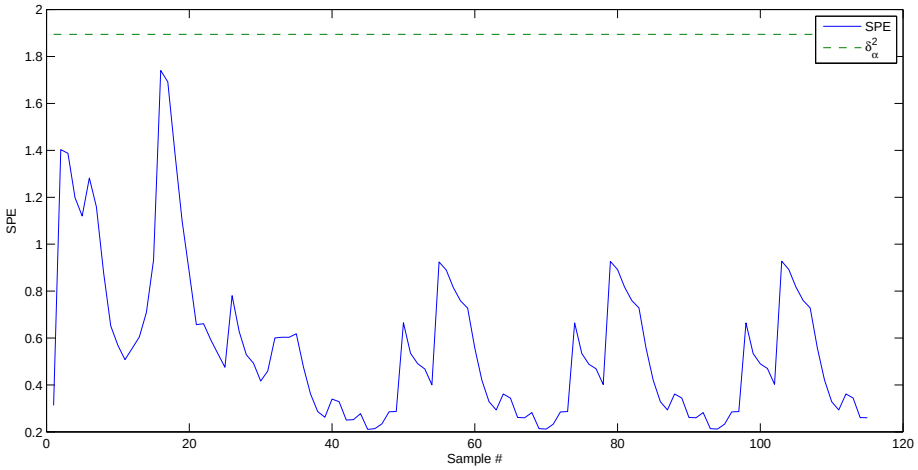


Fig. 6.2 SPE normal case

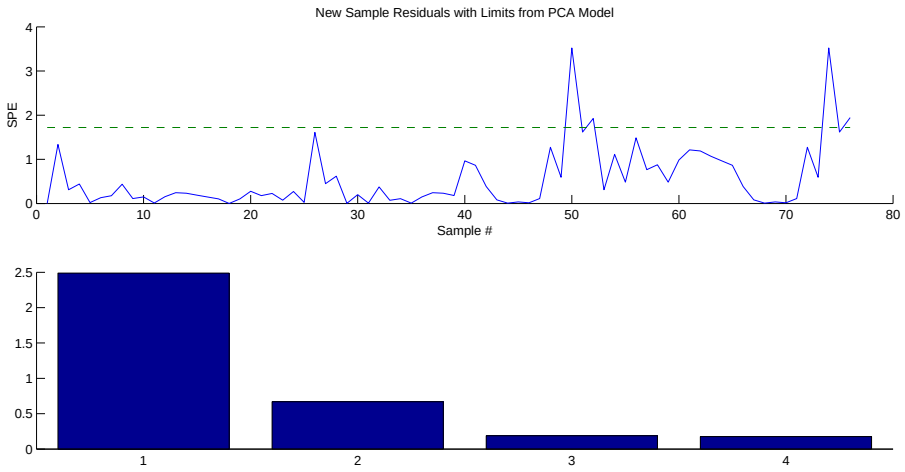


Fig. 6.3 (a)The statistical SPE of the global PCA for the abnormal state. (b)The contribution of the variables

To illustrate the practical utility of the Bottleneck Neural Network (BNN) technique in a real-world context of water distribution networks, we utilize the network exemplified in Figure 6.4 as a case illustration. This network encompasses pumping stations, a cascading dam, and three tanks. The monitoring focus of the network revolves around parameters like pressure, flow, and velocity linked to each individual tank.

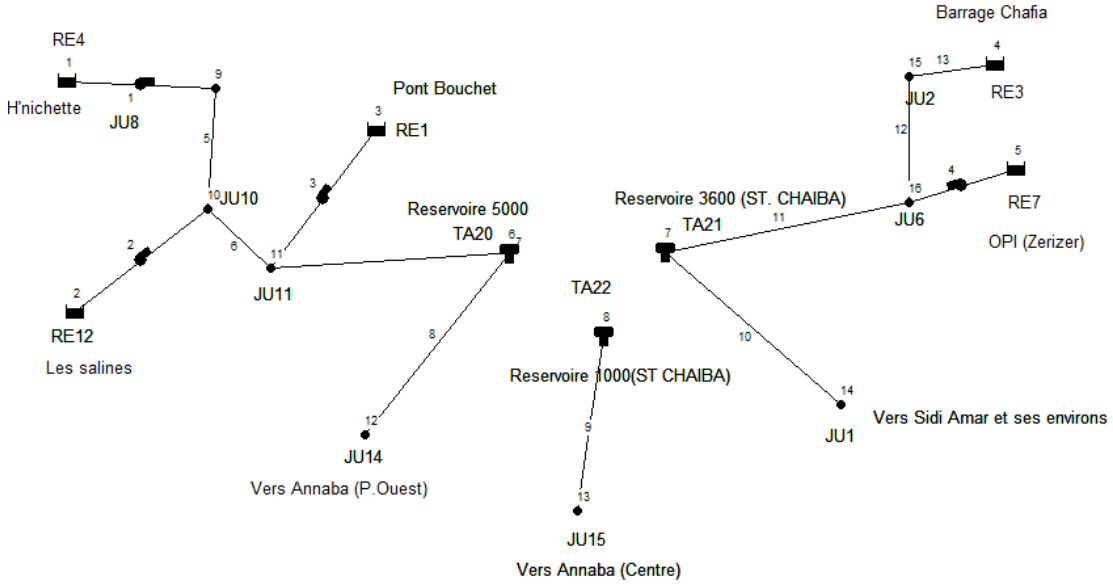


Fig. 6.4 The water distribution network under study

Let: $P_1(t), P_2(t), P_3(t)$: The pressure of the water discharged from each tank.

$q_1(t), q_2(t), q_3(t)$: The water flow originating from the tanks.

$V_1(t), V_2(t), V_3(t)$: The velocity measurement for each of the reservoirs under supervision.

Therfore, the input data are: $\mathrm{X} = [\mathrm{P}_1\ \mathrm{q}_1\ \mathrm{V}_1\ \mathrm{P}_2\ \mathrm{q}_2\ \mathrm{V}_2\ \mathrm{P}_3\ \mathrm{q}_3\ \mathrm{V}_3]$

Following the establishment of the hydraulic model, a fault is introduced into the system's dynamics starting from the $30^{th}$ sample and continuing until the simulation's end. As depicted in Figure 6.6 (a), the statistical behavior of the process variables, as measured by SPE, exhibits a discernible deviation starting from the $30^{th}$ sample onwards.

To identify the root cause of this fault, we analyze the contribution of each variable, as shown in Figure 6.6 (b). This analysis indicates that the flow sensor is responsible for the

fault occurring from the $30^{th}$ sample to the end of the simulation.

The results of the simulation vividly illustrate the utilization of the BNN approach in detecting anomalies in flow rates. This is achieved by observing the statistical behavior through SPE analysis and further understanding the fault's location via contribution plots.

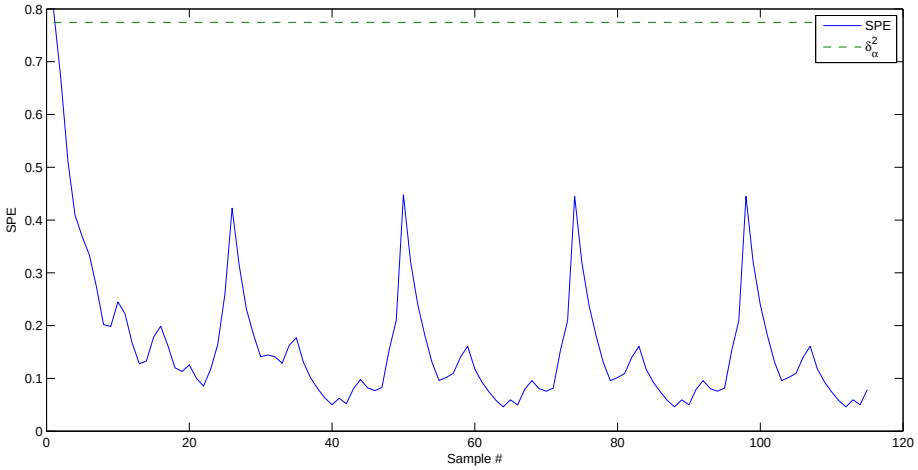


Fig. 6.5 SPE normal case

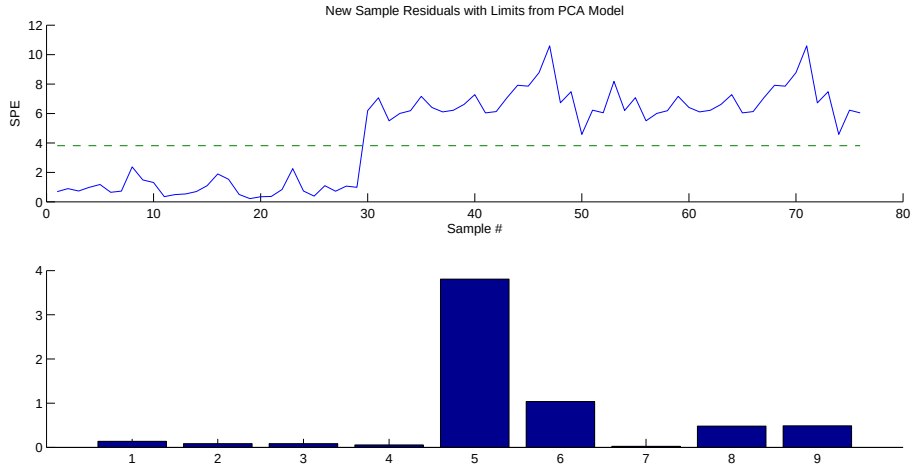


Fig. 6.6 (a)The statistical SPE of the global PCA for the abnormal state. (b)The contribution of the variables

## 6.2 Sensor Validation using BNN and Variable Reconstruction :Case Study

To showcase the efficacy of the proposed approach, a biological system is chosen as an illustrative example. This system involves a continuous aerated bioreactor utilized for wastewater treatment within the pulp and paper industry, as depicted in Figure6.7.

Within this bioreactor, a mixed microbial population thrives while metabolizing a combination of two distinct substrates: an energetic substrate and a xenobiotic substrate. It's noteworthy that the primary pollutant to address is the xenobiotic substrate, as the energetic one is more readily biodegradable, as explained in[47].

Let $c(t)[g/l]$: the biomass concentration.
$s(t)[g/l]$: the concentration of the xenobiotic pollutant substrate.
$e(t)[g/l]$: the concentration of the energetic substrate.
$u_1(t)[l/s]$: flow rate of clear water feeding the reactor.
$u_2(t)[l/s]$: flow rate of waste water. The waste water contains certain maximal substrate concentrations: $s_a^{max}[g/l]$ and $e_a^{max}[g/l]$.
$\mu_{sm}$ and $\mu_{em}$ : the growth rates of the substrates.
$Y_{c/s}$ and $Y_{c/e}$ : yield coefficients.
$K_s$ and $K_e$ : Michaelis-Menten constants.
$a_s$ and $a_e$ : Inhibiting coefficients.
The differential equations describing the behaviour of the process are:

$$\begin{cases} \dot{c}(t) = \mu_{sm}\left(\frac{s(t)}{K_s+s(t)+a_e e(t)} + \mu_{em}\frac{e(t)}{K_e+e(t)+a_s s(t)}\right)c(t) \\ \qquad\qquad -u_1(t)\,c(t) - u_2(t)\,c(t) \\ \dot{s}(t) = -\frac{\mu_{sm}}{Y_{c/s}}\frac{s(t)}{K_s+s(t)+a_e e(t)}c(t) - u_1(t)\,s(t) \\ \qquad\qquad +(s_a^{\max} - s(t))\,u_2(t) \\ \dot{e}(t) = -\frac{\mu_{em}}{Y_{c/e}}\frac{e(t)}{K_e+e(t)+a_s s(t)}c(t) - u_1(t)\,e(t) \\ \qquad\qquad +(e_a^{\max} - e(t))\,u_2(t) \end{cases}$$

The analytical model described earlier serves as a virtual system, generating data in an open-loop mode. The data collected under normal conditions is then utilized to construct the monitoring model based on competitive principal component analyzers. The data vector corresponding to the $k^{th}$ sample is expressed as follows:

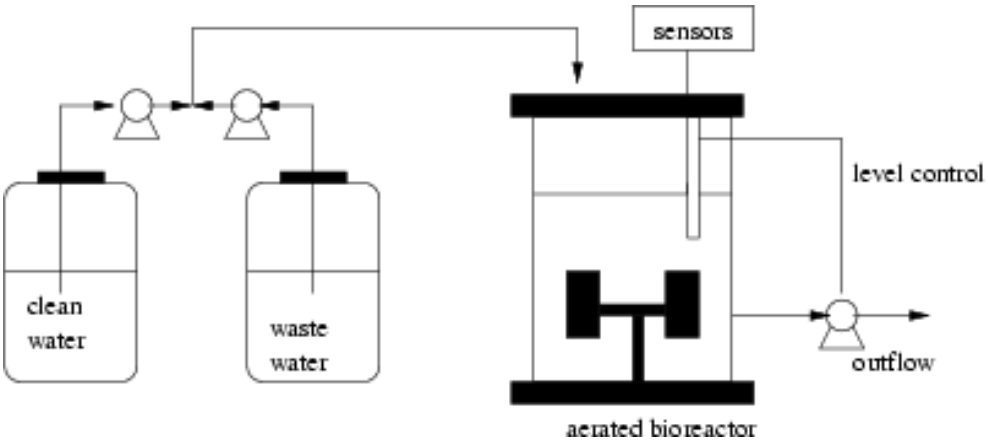


Fig. 6.7 Schematic representation of the aerated bioreactor for wastewater treatment

$$x(k) = \begin{bmatrix} c(k) & s(k) & e(k) & u_1(k) & u_2(k) \end{bmatrix}^T$$

The process of reconstructing the five variables under fault-free conditions is depicted in Figure6.8. Notably, each variable is reconstructed based on the information from the remaining sensors. The figure illustrates that despite the inherent nonlinearity of the process, the multiple PCA model successfully accomplishes the task of data reconstruction.

The proposed method is envisioned to serve as a valuable tool for the monitoring of multivariate signals.

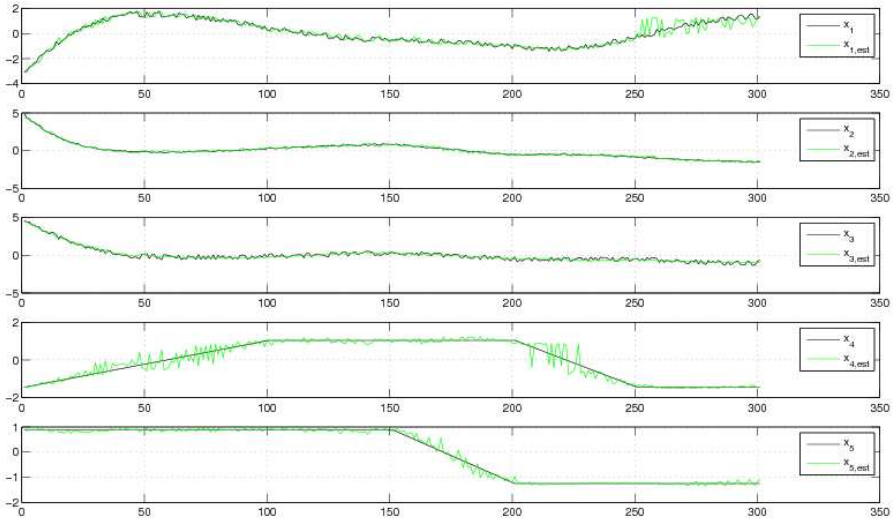

Fig. 6.8 The reconstruction of the variables in the fault free case. The variables $x_j$ and their reconstructed values $\hat{x}_j$, $j = 1,\ldots,5$.

Throughout our study, our primary focus has been directed towards sensor-related faults. The foundation of the BNN approach rests upon an auto-associative Neural Network model, comprising a total of five layers. The detection of sensor faults is accomplished through the application of the Squared Prediction Error (SPE) metric. This methodology is exemplified in Figure6.9, where the figure illustrates the BNN's capacity to effectively identify changes in the SPE across the simulation period.

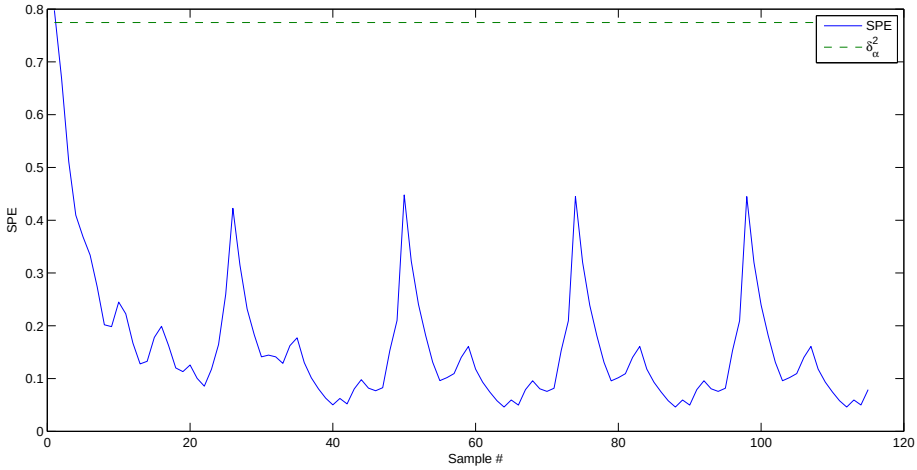


Fig. 6.9 SPE normal case

Figure6.10 displays the evolution of SPE, revealing two distinct operating regions. Notably, the second region commencing from the $30^{th}$ sample and extending to the simulation's conclusion signifies the presence of a sensor fault.

Following the development of the hydraulic model and within the simulation framework, a fault is deliberately introduced to the variable $e$, which represents the concentration of the energetic substrate. This fault occurs from the $30^{th}$ sample and extends until the simulation's conclusion. This behavior is visualized in Figure 6.10, which showcases the SPE statistical behavior of the process variables, indicating an offset starting from the $30^{th}$ sample.

Figure 6.11 further illustrates that the index of variable number 3 has been affected by a fault. This conclusion is drawn from the fact that this particular index is the only one that falls outside its control limit, as revealed by the monitoring approach.

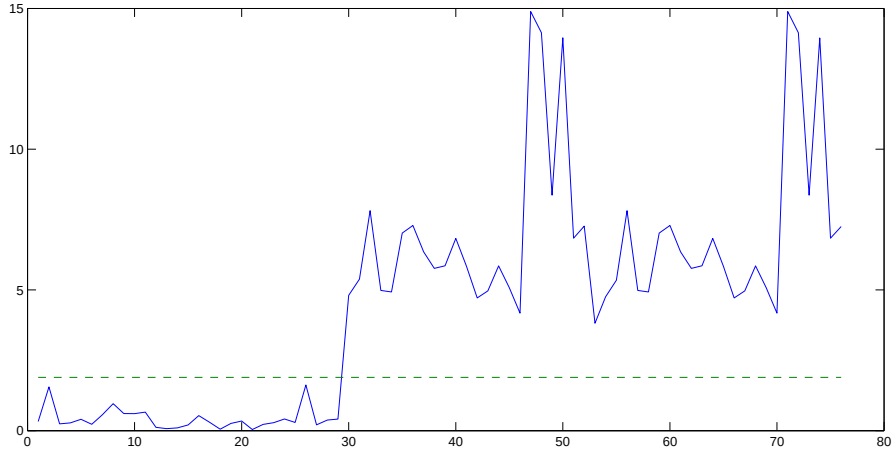


Fig. 6.10 SPE for faulty case

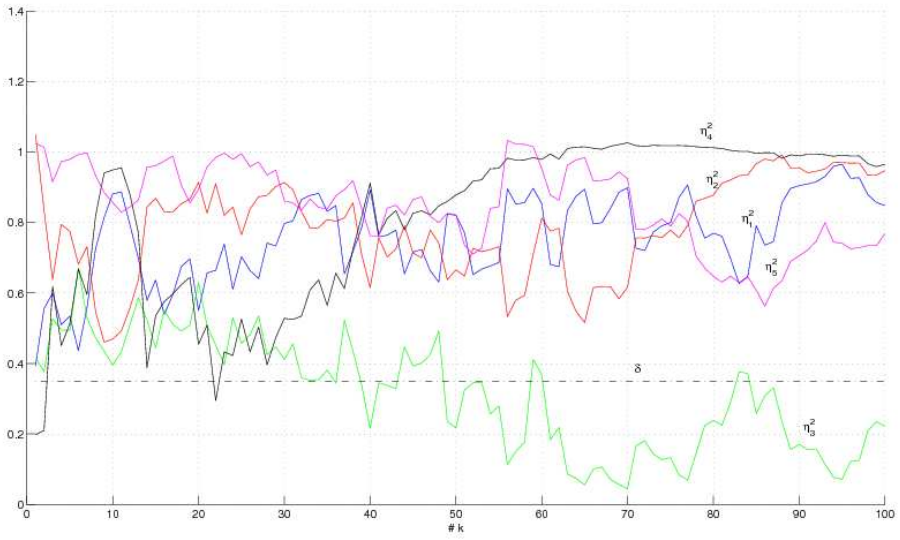


Fig. 6.11 SPE after reconstruction of variables

## 6.3 Fuzzy Predictive Control of Chlorine Concentration in DWDS :Case Study

Chlorine residual control of two DWDS is considered in the case study.

An evaluation benchmark involving two chlorine injection nodes and two monitored nodes is examined, as depicted in Figure6.12. This benchmark encompasses a network comprising 27 pipes, 16 nodes, and 3 reservoirs. The network operation involves two pumps, namely pump 201 and pump 101, responsible for transferring water from the sources, node 100 and node 200, across the entire network. Additionally, these pumps are also supplied by the tanks located at nodes 17, 18, and 19.

The chlorine concentration values measured at node 16 and node 8 correspond to the system's outputs, denoted as $y_1$ and $y_2$ respectively.
Regarding the input nodes denoted as $u_1$ and $u_2$, the chlorine concentration values at the injection nodes 5 and 10 are collected during the simulation steps.

The water demand is projected over a 55-hour horizon with intervals of 3 minutes. Figure 6.13 presents the time series data for both inputs and outputs.

The introduced T-S (Takagi-Sugeno) model encompasses two inputs and two outputs, specifically involving the chlorine concentration at node 16 and node 8. To evaluate the modeling outcomes, the Root Mean Square Error (RMSE) metric is utilized. RMSE is computed using the following formula:

$$RMSE = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left(\hat{y}(k) - y(k)\right)^2} \tag{6.5}$$

The assessment of model effectiveness is outlined in Table 6.3. The constructed model, distinguished by the presence of three local linear models ($c = 3$), yields Root Mean Square Error (RMSE) values of [0.0098, 0.0356] for node 16 and node 8 correspondingly. These values signify the model's adeptness in accurately capturing the underlying dynamics of the analyzed process.

Figure 6.14 illustrates the time evolution of the Takagi-Sugeno model over a $24-hour$ period in comparison with the process data. The figure underscores the model's accurate tracking capability. It's important to highlight that the model structure identification outlined in Chapter 3 is applicable to general complex nonlinear systems.

During the simulation, the predictive controller was designed using the following prediction parameters: prediction horizon $Hp = 12$, control horizon $Hc = 3$, and a sampling

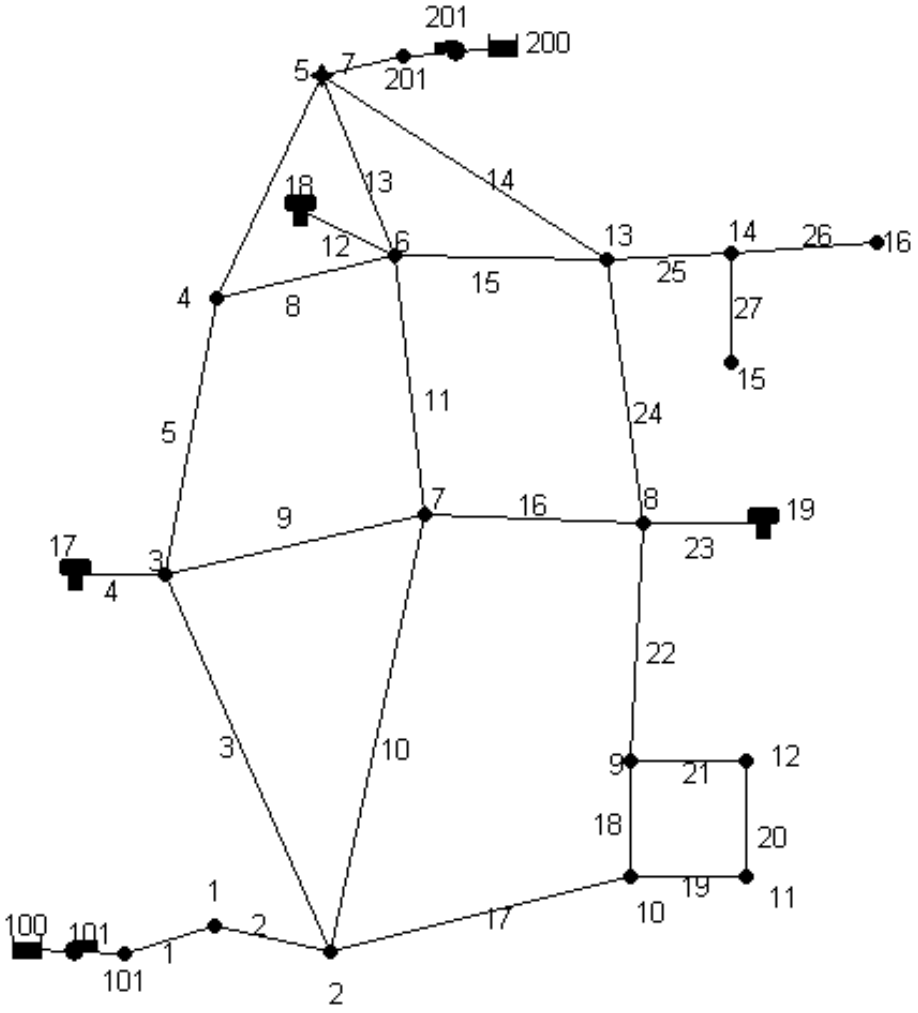


Fig. 6.12 A Benchmark Drinking Water Distribution Network

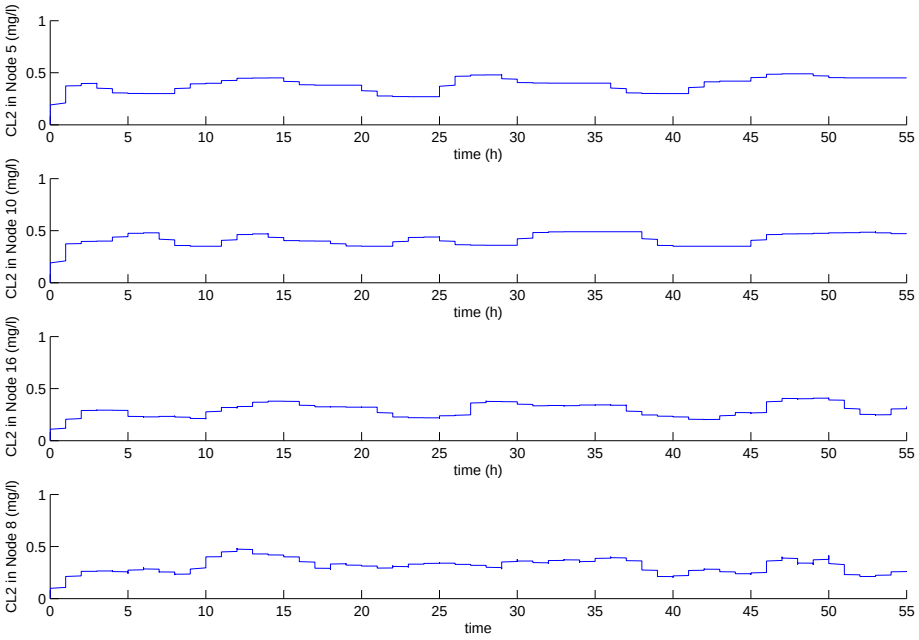


Fig. 6.13 Time series of chlorine injection and concentration

period $Ts = 3$ minutes.

Figure 6.15 visualizes both the control input signals directed towards the injection nodes and the performance of the Model Predictive Controller (MPC) to validate its capability for tracking setpoints. Notably, output constraints are established within the range of $0.1mg/l$ and $0.3mg/l$.

Table 6.3 T-S fuzzy model performances

| *Number of Clusters* | *RMSE node*16 | *RMSE node*8 |
|---|---|---|
| $C = 2$ | 0.0177 | 0.0756 |
| $C = 3$ | 0.0098 | 0.0365 |
| $C = 4$ | 0.0197 | 0.0945 |

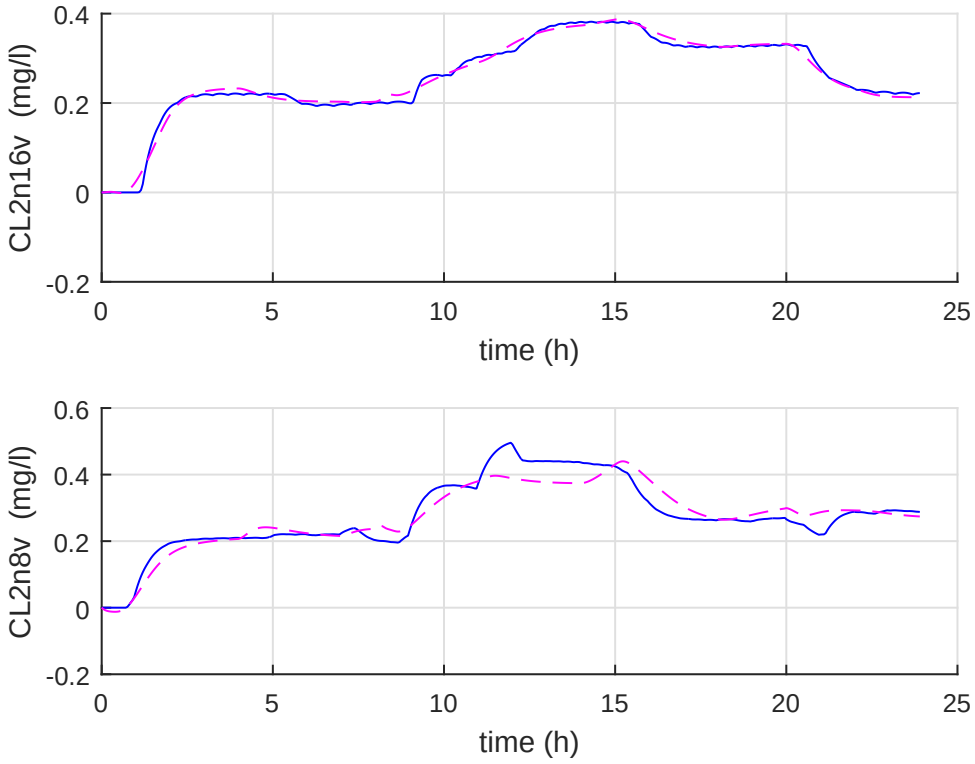


Fig. 6.14 Performance of the T-S fuzzy model of chlorine concentration in outputs nodes, $'-'$ data and$'--'$model estimation.

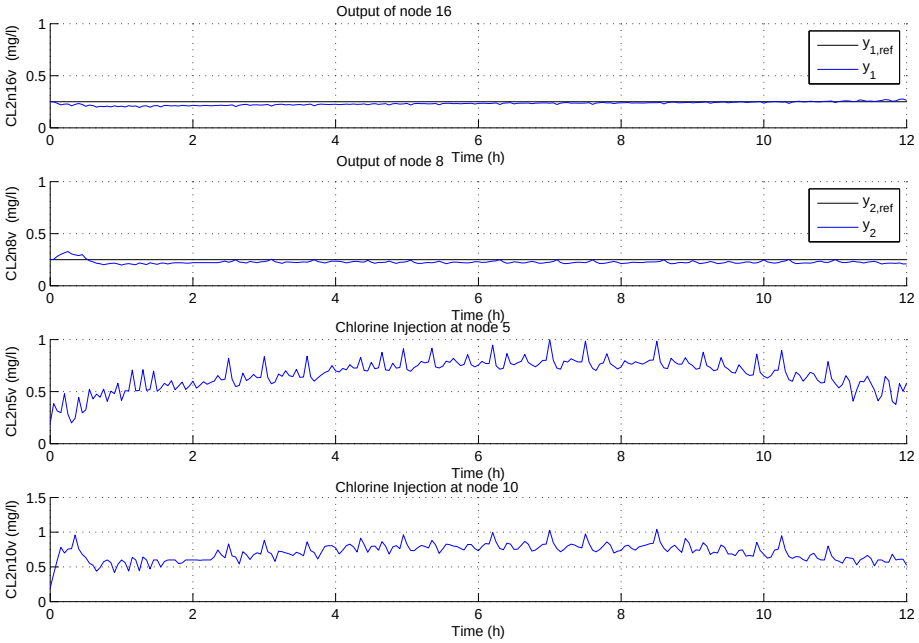


Fig. 6.15 Setpoint tracking performance of MPC based on T-S fuzzy model

### 6.3.1 Summary

This chapter introduces an innovative approach to detect and diagnose faults within systems characterized by nonlinear behavior. Our contribution involves replacing the linear PCA model with a Bottleneck Neural Network (BNN), thus enhancing the diagnostic method's adaptability to nonlinear systems.

The achieved results are regarded as satisfactory. By applying this approach to a quadruple-tank process, we effectively demonstrate its potential in detecting sensor faults. Additionally, we extend its applicability to real-world scenarios by employing it in an actual water distribution system.

In recent years, there has been significant development and application of fault detection and diagnosis methods aimed at enhancing process operations. Notably, the advancement of fault detection using Principal Component Analysis (PCA) has garnered attention due to its independence from prior knowledge about the process mechanism. The PCA's linear nature is expanded upon through the Nonlinear Principal Component Analysis (NLPCA) model, which utilizes a five-layer cascade neural network for nonlinear extension.

This part further reveals the successful application of constrained multivariable fuzzy predictive control in managing chlorine concentration within drinking water networks. This

control approach ensures that control signals are computed optimally for future process actions, focusing on achieving precise set-point regulation.

The construction of the Takagi-Sugeno fuzzy system follows two distinct steps: structure identification and parameter adaptation. The learning algorithm integrates self-adaptive learning rates based on the cost function's evolution. Simulation results validate the effectiveness of this proposed fuzzy control strategy in efficiently regulating chlorine concentration within drinking water distribution systems.

# Conclusion

The central aim of this thesis is to amalgamate advanced monitoring and control methodologies to effectively address both quantity and quality aspects within drinking water distribution systems. This integration is tailored to align with operational objectives, ensuring that consumers receive an appropriate supply of water in terms of both quantity and quality.

The research is predominantly structured around two core components. The initial part revolves around the diagnosis of anomalies and leaks in drinking water distribution networks. To achieve this, methodologies such as Non-linear Principal Component Analysis (NLPCA) and Bottleneck Neural Network (BNN) were employed. These techniques were utilized for the purpose of detecting, isolating, and pinpointing faults related to sensors within the system.

Utilizing the Bottleneck Neural Network encompasses the tasks of detecting and mitigating correlations among variables, thereby achieving dimensionality reduction, aiding in visualization, and facilitating exploratory data analysis. The prowess of the (BNN) lies in its capacity to unveil both linear and nonlinear correlations, independent of the specific characteristics of nonlinearities inherent in the dataset. The neural network's structure encompasses five layers: the input layer, mapping layer, bottleneck layer, de-mapping layer, and output layer. This configuration empowers the network to adeptly capture and portray the underlying relationships embedded within the data.

These neural networks belong to a distinct category of artificial neural networks that possess the capability to learn principal components directly, without the need to explicitly compute eigenvalues and eigenvectors from the sample covariance matrix. This feature streamlines the process of identifying the dominant patterns and relationships within the data.

The BNN methodology has been successfully applied through simulation on a benchmark system known as the quadruple-tank process. This application showcases the effectiveness of the method in detecting sensor faults within a controlled environment.

Furthermore, we extend the application of this method to a real water distribution system, highlighting its potential to identify sensor faults in practical scenarios. This demonstration underscores the method's capacity to detect anomalies in sensor data even in complex and real-world settings.

The second phase of this research pertains to controlling the chlorine concentration within the Drinking Water Distribution System (DWDS) while adhering to specified upper and lower limits. In this context, a constrained multivariable fuzzy predictive control strategy is successfully formulated. This approach involves predicting the future behavior of the physical process and using it to regulate the chlorine concentration.

The application of Fuzzy Model Predictive Control (FMPC) is aimed at satisfying output constraints. This is achieved through the construction of Takagi-Sugeno fuzzy models. The fuzzy model is derived from the collected input-output data using a tailored model identification algorithm. This algorithm incorporates fuzzy clustering to determine the antecedents of the fuzzy model rules. Additionally, the consequent parts are estimated through a least squares parameter estimation technique. This methodology ensures an effective control strategy while adhering to constraints and accounting for the uncertainty inherent in real-world processes.

The results and ensuing discussions are outlined in Chapter 6.

# References


[1] Abonyi, J. and Feil, B. (2007). *Cluster analysis for data mining and system identification*. Springer Science & Business Media.

[2] Alcala, C. F. and Qin, S. J. (2009). Reconstruction-based contribution for process monitoring. *Automatica*, 45(7):1593–1600.

[3] Azar, A. T. and Vaidyanathan, S. (2015). *Computational intelligence applications in modeling and control*. Springer.

[4] Babuška, R. and Kober, J. (2010). Knowledge-based control systems. *Delft University of Technology*.

[5] Bello, O., Hamam, Y., and Djouani, K. (2014). Fuzzy dynamic modelling and predictive control of a coagulation chemical dosing unit for water treatment plants. *Journal of Electrical Systems and Information Technology*, 1(2):129–143.

[6] Bouzenad, K., Ramdani, M., Zermi, N., and Mendaci, K. (2013). Use of nlpca for sensors fault detection and localization applied at wtp. In *2013 World Congress on Computer and Information Technology (WCCIT)*, pages 1–6. IEEE.

[7] Bouzid, S. and Ramdani, M. (2013a). Sensor fault detection and diagnosis in drinking water distribution networks. In *2013 8th International Workshop on Systems, Signal Processing and their Applications (WoSSPA)*, pages 378–383. IEEE.

[8] Bouzid, S. and Ramdani, M. (2013b). Water distribution system condition monitoring based on bottleneck neural networks. *Proceedings Engineering & Technology-Vol*, 3:112–116.

[9] Bouzid, S., Ramdani, M., and Chenikher, S. (2019). Quality fuzzy predictive control of water in drinking water systems. *Automatic Control and Computer Sciences*, 53(6):492–501.

[10] Brdys, M. and Chang, T. (2002). Robust model predictive control under output constraints. *IFAC Proceedings Volumes*, 35(1):41–46.

[11] Cao, J. and Gertler, J. (2004). Partial pca-based optimal structured residual design for fault isolation. In *Proceedings of the 2004 American Control Conference*, volume 5, pages 4420–4425. IEEE.

[12] caroline Cridelich, C. et al. (2015). *Influence of retraint systems during an automobile crash: prediction of injuries for frontal impact sled tests based on biomechanical data mining*. PhD thesis, Besançon.

[13] Chang, T. (2003). *Robust model predictive control of water quality in drinking water distribution systems*. PhD thesis, University of Birmingham.

[14] De Oliveira, J. V. and Pedrycz, W. (2007). *Advances in fuzzy clustering and its applications*. John Wiley & Sons.

[15] Dong, D. and McAvoy, T. J. (1996). Nonlinear principal component analysis—based on principal curves and neural networks. *Computers & Chemical Engineering*, 20(1):65–78.

[16] Dunia, R., Qin, S. J., Edgar, T. F., and McAvoy, T. J. (1996). Use of principal component analysis for sensor fault identification. *Computers & Chemical Engineering*, 20:S713–S718.

[17] Duzinkiewicz, K. (2006). Set membership estimation of parameters and variables in dynamic networks by recursive algorithms with a moving measurement window. *International Journal of Applied Mathematics and Computer Science*, 16(2):209–217.

[18] Eliades, D. G. and Polycarpou, M. M. (2007). Multi-objective optimization of water quality sensor placement in drinking water distribution networks. In *2007 European Control Conference (ECC)*, pages 1626–1633. IEEE.

[19] Gertler, J. (1999). Fault detection and diagnosis in engineering systems marcel dekker inc. *Nowy Jork*.

[20] Gertler, J. and Singer, D. (1990). A new structural framework for parity equation-based failure detection and isolation. *Automatica*, 26(2):381–388.

[21] Grüne, L. and Pannek, J. (2017). Nonlinear model predictive control. In *Nonlinear model predictive control*, pages 45–69. Springer.

[22] Hastie, T. and Stuetzle, W. (1989). Principal curves. *Journal of the American Statistical Association*, 84(406):502–516.

[23] Heloulou, N., Ramdani, M., and Abidi, A. (2014). Biological nitrogen removal process monitoring based on fuzzy robust pca. *Research Journal of Applied Sciences, Engineering and Technology*, 7(21):4434–4444.

[24] Hongxiang, W., Wenxian, G., Jianxin, X., and Hongmei, G. (2010). A hybrid pso for optimizing locations of booster chlorination stations in water distribution systems. In *2010 International Conference on Intelligent Computation Technology and Automation*, volume 1, pages 126–129. IEEE.

[25] Johansson, K. H. (2000). The quadruple-tank process: A multivariable laboratory process with an adjustable zero. *IEEE Transactions on control systems technology*, 8(3):456–465.

[26] Kanzawa, Y. and Miyamoto, S. (2018). Generalized fuzzy c-means clustering and its theoretical properties. In *Modeling Decisions for Artificial Intelligence: 15th International Conference, MDAI 2018, Mallorca, Spain, October 15–18, 2018, Proceedings 15*, pages 243–254. Springer.

[27] Kramer, M. A. (1992). Autoassociative neural networks. *Computers & chemical engineering*, 16(4):313–328.

[28] Łangowski, R. and Brdys, M. (2007). Monitoring of chlorine concentration in drinking water distribution systems using an interval estimator. *International Journal of Applied Mathematics and Computer Science*, 17(2):199–216.

[29] Łangowski, R. and Brdys, M. A. (2017). An interval estimator for chlorine monitoring in drinking water distribution systems under uncertain system dynamics, inputs and chlorine concentration measurement errors. *International Journal of Applied Mathematics and Computer Science*, 27(2):309–322.

[30] Liang, J. and Qian, J. (2003). Multivariate statistical process monitoring and control: Recent developments and applications to chemical industry. *Chinese Journal of Chemical Engineering*, 11(2):191–203.

[31] Ludwig, S. A. (2014). Clonal selection based fuzzy c-means algorithm for clustering. In *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*, pages 105–112.

[32] MacGregor, J. (1994). Statistical process control of multivariate processes. *IFAC Proceedings Volumes*, 27(2):427–437.

[33] MacGregor, J. F. and Kourti, T. (1995). Statistical process control of multivariate processes. *Control Engineering Practice*, 3(3):403–414.

[34] Maciejowski, J. M. (2002). *Predictive control: with constraints*. Pearson education.

[35] Mohammed, N. G. and Abdulrahman, A. (2009). Water supply network system control based on model predictive control. In *Proceedings of the International MultiConference of Engineers and Computer Scientists*, volume 2.

[36] Mollov, S., Babuska, R., Abonyi, J., and Verbruggen, H. B. (2004). Effective optimization for fuzzy model predictive control. *IEEE Transactions on fuzzy systems*, 12(5):661–675.

[37] Pardo, S. A. and Pardo, Y. A. (2016). *Empirical modeling and data analysis for engineers and applied scientists*. Springer.

[38] Qin, S. J. and McAvoy, T. J. (1992). Nonlinear pls modeling using neural networks. *Computers & Chemical Engineering*, 16(4):379–391.

[39] Raich, A. and Çinar, A. (1997). Diagnosis of process disturbances by statistical distance and angle measures. *Computers & chemical engineering*, 21(6):661–673.

[40] Rossman, L. A. et al. (1994). Epanet users manual.

[41] Shang, F., Uber, J. G., Rossman, L. A., and Janke, R. (2008). Epanet multi-species extension user's manual. *Risk Reduction Engineering Laboratory, US Environmental Protection Agency, Cincinnati, Ohio*.

[42] Shao, Y. E. and Hsu, B.-S. (2009). Determining the contributors for a multivariate spc chart signal using artificial neural networks and support vector machine. *International Journal of Innovative Computing, Information and Control*, 5(12):4899–4906.

[43] Stavroulakis, P. (2004). *Neuro-fuzzy and fuzzy-neural applications in telecommunications*. Springer.

[44] Su, C.-l. and Li, P. (2010). Adaptive predictive functional control based on takagi-sugeno model and its application to ph process. *Journal of Central South University of Technology*, 17(2):363–371.

[45] Sun, J. and Li, Y. (2016). Joint inversion of multiple geophysical and petrophysical data using generalized fuzzy clustering algorithms. *Geophysical Supplements to the Monthly Notices of the Royal Astronomical Society*, 208(2):1201–1216.

[46] Szczepaniak, P. S. and Lisboa, P. J. (2012). *Fuzzy systems in medicine*, volume 41. Physica.

[47] Visvanathan, C., Aim, R. B., and Parameshwaran, K. (2000). Membrane separation bioreactors for wastewater treatment. *Critical reviews in environmental science and technology*, 30(1):1–48.

[48] Webb, A. R. (1996). An approach to non-linear principal components analysis using radially symmetric kernel functions. *Statistics and computing*, 6(2):159–168.

[49] Wierzchoń, S. T. and Kłopotek, M. A. (2018). *Modern algorithms of cluster analysis*. Springer.

[50] Wilson, D. J., Irwin, G. W., and Lightbody, G. (1999). Rbf principal manifolds for process monitoring. *IEEE Transactions on Neural Networks*, 10(6):1424–1434.

[51] Xie, M. and Brdys, M. (2015). Nonlinear model predictive control of water quality in drinking water distribution systems with dbps objectives. *World Academy of Science, Engineering and Technology, International Journal of Mathematical, Computational, Physical, Electrical and Computer Engineering*, 9(8):457–463.

[52] Yan, S., Zhou, J., Zheng, Y., and Li, C. (2018). An improved hybrid backtracking search algorithm based t–s fuzzy model and its implementation to hydroelectric generating units. *Neurocomputing*, 275:2066–2079.

[53] Zamora, C., Giraldo, J. M., and Leirens, S. (2010). Model predictive control of water transportation networks. In *2010 IEEE ANDESCON*, pages 1–6. IEEE.