MODELING, MONITORING AND CONTROL OF REVERSE OSMOSIS DESALINATION PLANTS USING DATA-BASED TECHNIQUES

Xavier Pascual Caro

Dipòsit Legal: T 961-2014

MODELING, MONITORING AND CONTROL OF REVERSE OSMOSIS DESALINATION PLANTS USING DATA-BASED TECHNIQUES

DOCTORAL THESIS

Department of Chemical Engineering

Xavier Pascual Caro

Supervised by: Dr. Jaume Giralt Marcé

Dr. Robert Rallo Moya

UNIVERSITAT ROVIRA I VIRGILI

Tarragona

2014

UNIVERSITAT ROVIRA I VIRGILI

Departament d'Enginyeria Química
Campus Sescelades,
Avinguda Països Catalans, 26
43007 Tarragona
Tel: 977 55 86 75
Fax: 977 55 96 67

Dr. Jaume Giralt Marcé and Dr. Robert Rallo Moya,

CERTIFY that the present study, entitled "Modeling, monitoring and control of reverse osmosis desalination plants using data-based techniques" presented by Xavier Pascual Caro for the award of the degree of Doctor, has been carried out under our supervision at the Chemical Engineering Department of the University Rovira i Virgili.

Tarragona, 3rd February 2014,

Dr. Jaume Giralt Marcé

Dr. Robert Rallo Moya

Agraïments

M’agradaria agrair als meus dos directors de tesi, el Dr. Jaume Giralt i el Dr. Robert Rallo, per ser la meva guia durant tot aquest temps i per tots els vostres consells.

Al Dr. Yoram Cohen per l’oportunitat que m’ha donat de realitzar part del doctorat amb el seu grup de recerca i d’ampliar els meus coneixements al seu costat. També vull estendre aquests agraïments al Dr. Panagiotis D. Christofides i a tots els companys del grup de recerca PolySep.

Al Dr. Francesc Giralt i al Dr. Joan M. Ferrer pels seus consells i la seva ajuda de forma desinteressada que sobretot m’han aportat un esperit crític i un altre punt de vista a l’hora d’afrontar els reptes.

A tots els companys de laboratori d’aquests anys, que han viscut amb mi el dia a dia de tot aquest treball.

Al departament i a l’escola, i a tot el personal que m’ha donat un cop de mà sempre que ho he necessitat.

Als membres del tribunal per acceptar de judicar aquest treball.

I en especial, a la meva família que ha estat al meu costat donant-me suport en tot moment.

UNIVERSITAT ROVIRA I VIRGILI
MODELING, MONITORING AND CONTROL OF REVERSE OSMOSIS DESALINATION PLANTS USING DATA-BASED TECHNIQUES
Xavier Pascual Caro
Dipòsit Legal: T 961-2014

Resum

Aquesta tesi proposa i avalua diverses aplicacions del modelat basat en dades de sistemes de dessalinització d'aigua mitjançant osmosi inversa per a la millora i optimització de la seva operació. El treball dut a terme durant aquests estudis ha estat basat en les dades obtingudes a partir d'experiments realitzats a la planta pilot d'osmosi inversa Mini-Mobile-Modular (M3), dissenyada a la UCLA per a la producció d'aigua potable. Les dades obtingudes han estat tractades i normalitzades per al seu ús en el desenvolupament dels models de la planta, els quals han estat generats utilitzant la tècnica de Support Vector Regression (SVR).

En primer lloc, s'han desenvolupat models predictius del comportament de la planta M3 operant en estat estacionari. Un cop fet el procés de selecció de variables d'entrada per als models, s'ha demostrat que aquests són capaços de predir amb un nivell de precisió molt elevat, fins i tot superior a tècniques basades en primers principis, totes les variables de sortida de la planta, utilitzant com a paràmetres del model únicament les variables d'entrada de la planta. Addicionalment, s'ha comprovat que el desenvolupament d'un model capaç de predir totes les variables de sortida de la planta simultàniament, proporciona un nivell de precisió comparable al dels models desenvolupats individualment per a cadascuna de les variables. Per tant, aquest model global, és ideal per a una predicció ràpida i acurada del funcionament de tota la planta.

En segon lloc, també s'han desenvolupat models de predicció a curt termini de la conductivitat de l'aigua produïda per la planta durant operació en estat no estacionari (operació dinàmica). Degut a la complexitat inherent de l'operació dinàmica, tot el conjunt de dades experimentals s'ha classificat mitjançant l'ús de Mapes Autoorganitzats (SOM). D'aquesta manera, s'aconsegueix seleccionar el conjunt de dades òptim per a la generació dels models i així assegurar que els diferents tipus d'operació dinàmica de la planta queden coberts a l'entrenament (generació) dels models. L'alta precisió de les prediccions obtingudes, suggereix que els models basats en dades de la planta poden ser de gran utilitat

aplicats a sistemes de control avançat de plantes d'osmosi inversa, així com en sistemes de detecció de fallades i optimització de processos.

Finalment, s'han proposat i comparat dues eines basades en dades per a la detecció de fallades, isolació, correcció i imputació de dades per als sensors de la planta pilot M3. L'eina basada en Mapes Autoorganitzats (SOM), ha demostrat ser capaç de detectar, imputar i corregir amb gran precisió les errades que ocorren a la planta, amés de proporcionar una eina visual per a la detecció ràpida de fallades en l'operació de la planta. Per altra banda, l'eina basada en Support Vector Regression (SVR) millora tant el nivell de detecció com la precisió en la correcció i imputació de les dades. Aquest alt nivell de detecció i precisió, permet la detecció de fallades en sensors fins i tot quan les desviacions respecte els valors esperats d'aquests són molt petites.

En general, aquesta tesi presenta un marc de treball de modelització de processos d'osmosi inversa basat en dades que millora significativament els resultats obtinguts amb mètodes tradicionals com els models basats en primers principis.

Summary

This thesis proposes and assesses several data-driven modeling applications for water desalination reverse osmosis systems to improve and optimize its operation. The present work is based on data obtained from experiments carried out in the Mini-Mobile-Modular (M3) pilot plant, designed at UCLA for the production of fresh water. The data have been treated and normalized to develop plant models, which have been generated using the Support Vector Regression (SVR) technique.

First, steady state models have been developed for the prediction of the Mini-Mobile-Modular pilot plant (M3) operation, prior implementation of a variable selection process for the input model variables. It has been demonstrated that the steady state models developed are able to predict all the plant output variables using only plant feed conditions as inputs for the models. The accuracy achieved with this models has been even higher than first principle based techniques In addition, it has been demonstrated that a single composite model predicting simultaneously all the output variables of the plant performs with a comparable accuracy to the single output variable models. Therefore, the composite model developed is ideal for a quick and accurate prediction of the overall plant behavior.

Secondly, short-term prediction models for the conductivity of the fresh water produced have been developed for the plant operating under non-steady state conditions (dynamic operation). Due to the inherent complexity of dynamic operation, the whole experimental dataset has been classified using Self-Organizing Maps (SOM). Thereby one can select the optimal set of data for the generation of the models and thus assure that all possible dynamic operation modes of the plant are covered during the model training (generation). The high level of accuracy obtained by the predictions of the models, suggests that the data-driven models could be useful for advanced reverse osmosis plant control algorithms, fault detection and process optimization.

Finally, two data-driven tools have been proposed and compared for fault detection, isolation, and data imputation and correction of the M3 plant sensors. One of the tools, developed using Self-Organizing Maps, has shown to be capable of detecting and correcting with a high level of accuracy faults occurring in the plant, as well as providing a useful visual tool for the rapid fault detection during plant operation. On the other hand, the tool based on Support Vector Regression provides better detection levels and more accurate corrections of the faulty sensor data. This high level of detection and accuracy enables the detection of sensor faults even when the deviations with respect to the expected values are very small.

Overall, this thesis presents a modeling framework for reverse osmosis processes based on plant data, which improves significantly the performance obtained with other traditional methods such as the ones based on first principles.

Resumen

Esta tesis propone y evalúa diversas aplicaciones del modelado basado en datos de sistemas de desalación de agua mediante ósmosis inversa para la mejora y optimización de su operación. El trabajo llevado a cabo durante estos estudios ha estado basado en datos obtenidos a partir de experimentos realizados en la planta piloto de ósmosis inversa Mini-Mobile-Modular (M3), diseñada en UCLA para la producción de agua potable. Los datos obtenidos han sido tratados y normalizados para su posterior uso en el desarrollo de modelos de la planta, los cuales han sido generados utilizando la técnica de Support Vector Regression (SVR).

En primer lugar, se han desarrollado modelos predictivos del comportamiento de la planta M3 operando en estado estacionario. Una vez concluido el proceso de selección de variables de entrada para los modelos, se ha demostrado que éstos son capaces de predecir con un nivel de precisión muy elevado, incluso superior a técnicas basadas en primeros principios, todas las variables de salida de la planta, utilizando como parámetros de los modelos únicamente las variables de entrada de la planta. Adicionalmente, se ha comprobado que el desarrollo de un modelo capaz de predecir todas las variables de salida de la planta simultáneamente, proporciona un nivel de precisión comparable al de los modelos desarrollados individualmente para a cada una de las variables. Por lo tanto, este modelo global, es ideal para una predicción rápida y precisa del funcionamiento de toda la planta.

En segundo lugar, también se han desarrollado modelos de predicción a corto plazo de la conductividad del agua producida por la planta durante la operación en estado no estacionario (operación dinámica). Debido a la complejidad inherente de la operación dinámica, todo el conjunto de datos experimentales se ha clasificado mediante el uso de Mapas Autoorganizados (SOM). De esta forma, se consigue seleccionar el conjunto óptimo de datos para la generación de los modelos y así asegurar que los diferentes tipos de operación dinámica de la planta quedan cubiertos en el entrenamiento (generación) de los

modelos. La alta precisión de las predicciones obtenidas, sugiere que los modelos basados en datos de la planta pueden ser de gran utilidad aplicados a sistemas de control avanzado de plantas de ósmosis inversa, así como en sistemas de detección de fallos y optimización de procesos.

Finalmente, se han propuesto y comparado dos herramientas basadas en datos para la detección de fallos, aislamiento, corrección e imputación de datos para los sensores de la planta piloto M3. La herramienta basada en Mapas Autoorganizados (SOM), ha demostrado ser capaz de detectar, imputar y corregir con gran precisión los errores ocurridos en la planta, además de proporcionar una herramienta visual para la rápida detección de fallos durante la operación de la planta. Por otro lado, la herramienta basada en Support Vector Regression (SVR) mejora tanto el nivel de detección como la precisión en la corrección e imputación de los datos. Este alto nivel de detección y precisión, permite la detección de fallos en sensores incluso cuando las desviaciones respecto los valores esperados de éstos son muy pequeñas.

En general, esta tesis presenta un marco de trabajo de modelización de procesos de ósmosis inversa basado en datos que mejora significativamente los resultados obtenidos con métodos tradicionales como los modelos basados en primeros principios.

Table of contents

1. Objectives

The objective of this study is to develop accurate data-driven models (Support Vector Regression Models) able to describe the behavior of a reverse osmosis (RO) water desalination pilot plant during steady state and dynamic operation, under changing operating conditions over its entire operational domain (given by the manufacturer's safety recommendations). In addition, two sensor fault detection and isolation systems based on data-driven models (Self-Organizing Maps and Support Vector Regression) are compared and demonstrated for the RO pilot plant. The sensor fault detection and isolation systems are capable of reconstructing faulty data and carrying out data imputation for missing sensor data.

2. Organization

The work accomplished in this thesis is presented in two publications. The first article entitled "Data-driven models of steady state and transient operations of spiral-wound RO plant" proposes an approach, which uses data-driven models developed using the support vector regression (SVR) technique for the prediction of the behavior of a reverse osmosis water desalination pilot plant (M3). The approach has been demonstrated by simulating the plant operation under steady state and dynamic regimes. The steady state approach enabled the prediction of all the plant output variables with relative errors with respect to the expected values smaller than 2.5%, using only the input variables of the plant as parameters for the models. The dynamic operation of the plant has been described by short-term prediction models which track the evolution of the conductivity of the produced fresh water under changing plant conditions. The models have achieved performance levels of less than 1% absolute average relative errors for forecasting times of 2 s to 3.5 min.

The second article, entitled "Fault detection and isolation in spiral-wound RO desalination plant", presents two approaches for the fault detection and isolation applied to the M3 plant, as well as data reconciliation and data imputation mechanisms. The first approach is based on support vector regression models that simulate the M3 plant sensors behavior and perform predictions and data reconstructions with absolute average relative errors smaller

than 1%. This level of accuracy enables the detection of faults even smaller than 5% with respect to the expected sensor values operating under normal conditions. The detection levels achieved for faults with deviations from ±4% to ±9% is greater than 91%, and for faults with deviations greater than ±10%, the detection levels have been 100%. On the other hand, the SOM-based approach does not achieve the same performance levels as the SVR-based one. The data reconstruction of the faulty sensors has been achieved with absolute average relative errors smaller than 5%. Due to the model accuracy, the detection of faults smaller than ±10% has not performed acceptable levels. However, faults with deviations of ±10% or greater have obtained false positives levels of 1.54% and levels of 6.64% false negatives. Nevertheless, the SOM-based approach provides a visual aid for a rapid fault detection as well as data reconciliation.

Publications

1. X. Pascual, H, Gu, A. Bartman, A. Zhu, A. Rahardianto, J. Giralt, R. Rallo, P.D. Christofides and Y. Cohen; Data-driven models of steady state and transient operations of spiral-wound RO plant, Desalination 2013, 316, 154-161.

2. X. Pascual, H, Gu, A. Bartman, A. Zhu, A. Rahardianto, J. Giralt, R. Rallo, P.D. Christofides and Y. Cohen; Fault detection and isolation in a spiral-wound reverse osmosis (RO) desalination plant, Ind. Eng. Chem. Res. In press.

Conference proceedings

1. X. Pascual, H, Gu, A. Bartman, A. Rahardianto, A. Zhu, J. Giralt, R. Rallo, and Y. Cohen, "Rapid Prototyping of Reverse Osmosis Processes Using Data-Driven Models" AIChE Annual Meeting, paper, Salt Lake City, Utah, 2010.

2. X. Pascual, H, Gu, A. Bartman, A. Zhu, A. Rahardianto, J. Giralt, R. Rallo, F. Giralt and Y. Cohen, "Diagnostic analysis and fault detection of RO membrane desalination via machine learning" ICOM 2011, paper, Amsterdam, The Netherlands, 2011.

3. X. Pascual, H, Gu, A. Bartman, A. Zhu, A. Rahardianto, J. Giralt, R. Rallo, J. Ferrer-Giné, F. Giralt and Y. Cohen, "Dynamic Data-Driven Models of Reverse Osmosis Desalination Systems" AIChE Annual Meeting, paper, Minneapolis, Minnesota, 2011.

4. X. Pascual, H, Gu, A. Bartman, A. Zhu, A. Rahardianto, J. Giralt, R. Rallo, P. D. Christofides and Y. Cohen, "RO Plant Performance Predictions and Fault Detection

with Data-Driven Models" Desalination Using Membrane Technology, paper, Sitges, Spain, 2013.

5. H. Gu, L. Gao, X. Pascual, A. Rahardianto, P. D. Christofides and Y. Cohen, "Novel System and Field Characterization of Self-Adaptive UF-RO Membrane Desalination of Coastal Seawater" AIChE Annual Meeting, paper, San Francisco, California, 2013.

3. Introduction

Over 70% of Earth's surface is covered by water, although less than 3% is freshwater (water with less than 1000 mg of dissolved solids per liter), and less than 1% of it (~0.007% of all water on Earth) is accessible for direct human uses in rivers, lakes reservoirs and shallow underground sources (Gleick 2011). Increasing drought conditions and demand of fresh water suitable for human uses make necessary the generation of new water sources such as seawater and brackish water as well as water reuse. As 97% of Earth's water supply is held within the oceans, seawater desalination has become one of the best alternatives. One of the most cost-efficient technologies in this area is reverse osmosis (RO), since it is able to achieve high removals of constituents while is not so energy intensive compared to other desalination processes.

3.1 **Reverse osmosis**

Reverse osmosis is a pressure-driven membrane filtration process used for removing low molecular weight solutes, such as inorganic salts and small organic molecules from a solvent. A semi permeable membrane is used to retain solute molecules, while solvent molecules are allowed to pass through it. The process called osmosis occurs when two solutions of different concentrations are separated by a semi permeable membrane, and the solvent from the lower concentration solution flows through the membrane into the concentrated one. The observed flow is due to the tendency to equalize the concentrations of both sides. Nevertheless, when in one side of the membrane the solution is composed by a pure solvent, the concentrations of both sides can never equal. In this case, the osmotic flow proceeds until the equilibrium between the chemical potentials of both sides is reached, which occurs when the pressure exerted by the concentrated solution against the membrane is high enough to prevent any further solvent flow. The hydrostatic pressure difference between the two sides of the membrane at this equilibrium point is known as osmotic pressure. In the process known as reverse osmosis, pressure is applied in the concentrated solution side to surpass the osmotic pressure and generate a flux of solvent through the membrane against the concentration gradient, as shown in Fig. 1.

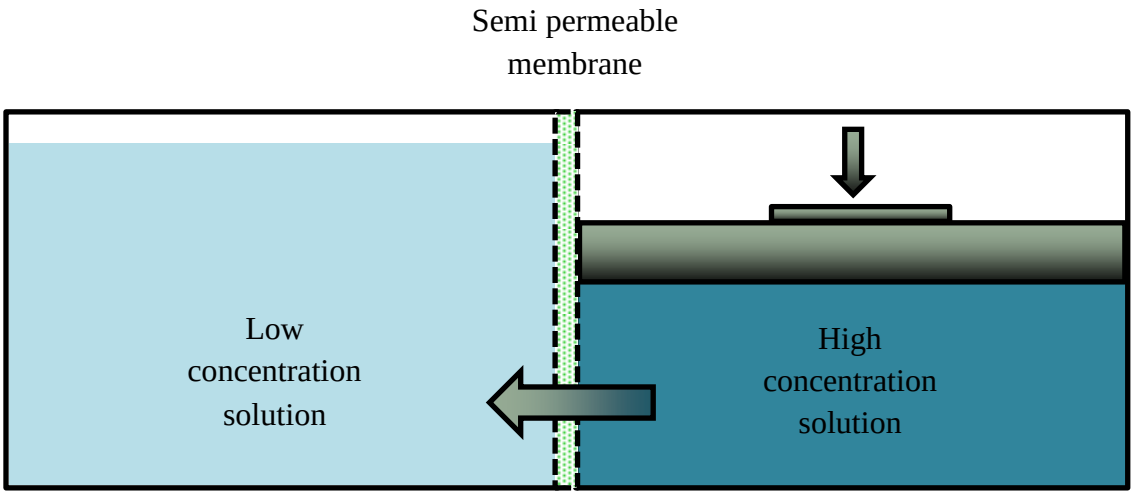

Fig. 1. Reverse osmosis process schematic.

The permeation of solvent through the membrane is driven by pressure (convection), and can be expressed by the following model:

$$J_w = A\left(\Delta P - \sigma \Delta \pi\right), \tag{1}$$

where J_w is the mass flux that passes through the membrane [kg · m^{2} · s^{-1}], A is the permeability coefficient (specific for each membrane) [kg · m^{-2} · kPa^{-1} · s^{-1}], ΔP is the pressure difference between the two sides of the membrane [kPa], $\Delta \pi$ is the osmotic pressure difference, and σ is the reflection coefficient, which measures the membrane selectivity (i.e., σ =0, no membrane selectivity; 0< σ <1, not completely semi permeable with solute transport; σ =1, ideal membrane without solute transport).

Differently the mass transfer of the solutes is controlled by diffusion, being the gradient of concentrations the driving force. The mass flux of solutes through the membrane can be described as:

$$J_s = B\left(C_f - C_p\right), \tag{2}$$

where J_s is the mass flux of solute through the membrane [kg · m^{2} · s^{-1}], B is the solute permeability [m · s^{-1}], and C_f and C_p are the concentrations of the feed and permeate

Dipòsit Legal: T 961-2014

solutions respectively [kg · m^3].The solute permeability B can be expressed as:

$$B = \frac{D_s K_s}{l}, \tag{3}$$

where D_s is the diffusion coefficient for the solute in the membrane [$m^2 \cdot s^{-1}$], K_s is the partition coefficient of the solute, l is the membrane thickness [m]. Thereby, the total permeate volumetric flux can be obtained by summing the solute and solvent fluxes:

$$J_p = \frac{\left(J_w + J_s\right)}{\rho_p} \tag{4}$$

where ρ_p is the permeate density [kg · m^3].

The performance of a reverse osmosis process can be expressed in terms of solvent recovery or solute rejection. The recovery (Y) is the fraction of solvent feed flow that passes through the membrane (Eq. 1.5), while rejection (R) is defined as the fraction of solute rejected by the membrane (Eq. 1.6).

$$Y = \frac{Q_p}{Q_f} \tag{5}$$

$$R = 1 - \frac{C_p}{C_f} \tag{6}$$

where Q_p and Q_f are the permeate and feed volumetric flow rates respectively [$m^3 \cdot s^{-1}$], and C_p and C_f are the permeate and feed concentrations respectively [kg · m^{-3}].

One of the most influencing parameters on the separation efficiency is the membrane material. In order to obtain a high separation, the flux of the solvent through the membrane needs to be maximized, while on the contrary, the solute flux needs to be minimized. This selectivity can be obtained by selecting a membrane material with high affinity for the solvent, and an affinity for the solute as low as possible. Two materials make the bulk of commercial reverse osmosis membranes, cellulose acetate (CA) and aromatic polyamide (PA). Cellulose acetate membranes perform higher fluxes requiring smaller membrane areas. They are resistant to small concentrations of free chlorine and may therefore be kept

free of bacteria and also produce a product with residual chlorine in it to prevent subsequent re-growth. On the other hand, polyamide membranes can be used at higher temperatures (up to 35ºC) than cellulose acetate membranes (up to 30ºC), they cannot tolerate chlorine, but they are not attacked by bacteria, whereas some bacteria which can occur in surface water in woodlands actually destroy cellulose acetate. Finally, polyamides can be used over a much wider pH range (4-11) than cellulose acetate (4-7.5). Each of the two materials has advantages and disadvantages; therefore the choice of the membrane material depends upon the nature of the input of water.

The configuration of the membrane has also an important impact in the separation process performance. Spiral wound membranes in cross flow configuration are one of the most used types of membranes in water desalination, since they offer high surface area per unit volume leading to high permeate fluxes. Despite its cylindrical configuration, this kind of membrane is essentially a flat-sheet device rolled with separating spacer mats. The feed solution passes through the module axially, while permeate moves in the spiral, radial direction toward the central permeate collection tube (Fig. 2).

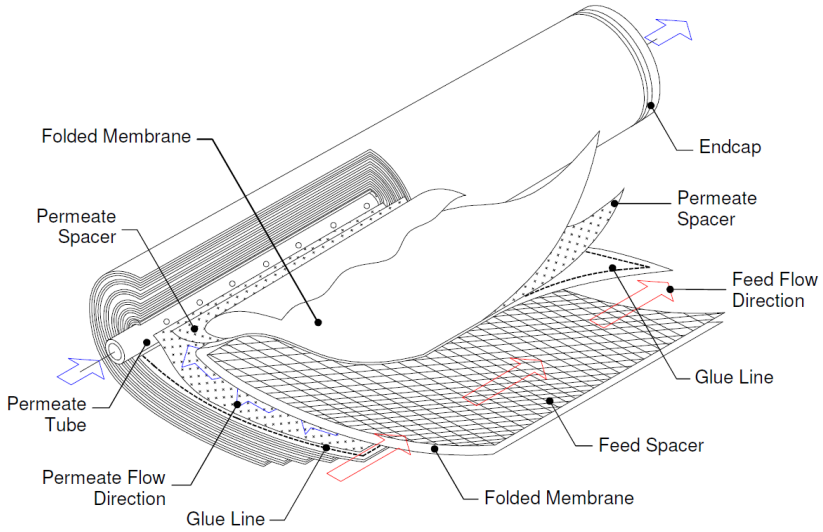

Figure 2. Spiral wound RO membrane module (Johnson and Busch 2010).

3.2 **Concentration polarization, fouling and scaling**

Although reverse osmosis is one of the most cost efficient technologies to produce purified water, there are some problems that can occur during plant operation that reduce the separation efficiency and membrane life-time. One of these phenomena is concentration polarization, which consists on the formation of a high concentration layer of dissolved and suspended particles near the membrane surface. When the feed water comes in contact with the membrane surface, water diffuses rapidly through the membrane, but the dissolved and suspended particles are either retained or they diffuse at lower rate than water, so they remain at the surface of the feed water side of the membrane. Solids accumulate to concentrations that exceed their concentrations in the bulk of the feed water. Concentration polarization can be deduced from a film theory model (Eq. 1.7).

$$\frac{C_m - C_p}{C_b - C_p} = \exp\left(\frac{J_p}{k}\right) \tag{7}$$

where, C is the concentration [kg · m^{-3}], subscripts m, p and b denote membrane surface, permeate flow and bulk solution respectively, J_p is the permeate flux [$m \cdot s^{-1}$] and k is the mass transfer coefficient [$m \cdot s^{-1}$].

If the concentrations of the dissolved solids exceed their solubility, they may precipitate and form a mineral scaling layer. Scalants are chemicals such as calcium carbonate, calcium sulfate, strontium sulfate, and reactive silica that are difficult to remove from RO membranes due to their low solubility. Scaling produces additional solids within the feed water and increases cleaning frequency. Scale on membranes produces nucleation sites, which increases the rate of additional scale formation. There are two main pretreatment methods to prevent scaling. The first group of techniques is based on ion exchange which removes scale-forming species from the feed water, while the other techniques are based on the addition of chemicals that change the characteristics of the feed water.

Membrane fouling occurs when suspended solids, microbes or organic material deposit on the membrane surface and form layers of undesired material that can plug the membrane and reduce its permeability. Fouling is also promoted by concentration polarization, since it

provides high concentrations of nutrients needed by bacteria for their growth. Colloids and organics are also concentrated in the membrane surface. The two main techniques to prevent fouling are mechanical processes that physically remove the suspended solids and chemical treatments that deactivate the foulants.

Apart from fouling and scaling, membrane degradation can also affect the RO process performance. It occurs when membranes are exposed to conditions that destroy the polymers they are made of. Some membrane materials can be hydrolyzed at high and low pH, while others can be degraded when exposed to oxidizers such as chlorine. The main prevention technique for membrane degradation due to acidic or alkaline waters is the addition of acid or base chemicals into the feed water to correct the pH and make the solution nearly neutral. Dechlorination is used to prevent oxidation of the membranes (Kucera 1997).

Beside the mentioned techniques used to mitigate fouling and scaling, there are some methods used to reduce the concentration polarization. The most used ones are increasing flow rate, impulse and agitating methods, intensifying the turbulent flow, periodic depressurization of the membrane tube, flow reversal, precoating the membrane surfaces and modification of membrane chemistry and surface.

3.3 **Alternatives to reverse osmosis**

Although RO is mainly used for water treatment, its application extends to other fields such as food and dairy industry, car washing, pharmaceutical and cosmetic production and reef aquariums among others. In addition to RO, there are several desalination technologies such as distillation and electrodialisys able to produce fresh water able for human use from seawater. The main drawback of these techniques is the high energy requirements. Electrodialisys, is more economic when the salinity of the feed water is low (brackish waters with less than 3000 ppm of dissolved solids), since its energy requirements are proportional to the water's salinity. Accordingly, it is an expensive technology for seawater desalination. The most used distillation techniques for water desalination are multi-stage flash (MSF) and multiple effect distillation (MED). The main advantages of these techniques are that they are less prone to fouling and scaling, and they do not require as

much pretreatment as RO does. Although its efficiency has increased during the last years, the requirement of high temperatures makes the energy consumption much greater than RO, which operates at ambient temperature. Even though RO has also high energy consumption due to the high pressures required, it can be recovered using pressure exchangers or energy recovery devices.

3.4 **Desalination in Spain**

The first seawater desalination plant in Spain was installed in 1964 in Lanzarote, a MSF plant with a capacity of 2500 m^3/day. Since then, a growing number of desalination plants have been installed, especially in the Mediterranean coast and the Canary Islands to such an extent that currently Spain's desalted water capacity is 8% of the global desalination capacity, while its population is only 0.6% of the world population. According to the Global Water Intelligence and the International Desalination Association, only Saudi Arabia (17%), United Arab Emirates (UAE) (13%) and USA (13%) are ahead in the production of desalted water. Such an increase has been deeply affected by the political situation in Spain. In 2001, in order to mitigate the poor hydrologic situation of the country, a new proposal of the Plan Hidrológico Nacional (National Hydrologic Plant) was created, when the right-wing party PP was in the government. Among other actions, it promoted the transfer of 1050 hm^3/year of water from the Lower Ebro River in Catalonia, to the arid areas in the southeast of the peninsula (PHN 2001). After the 2004 national elections, the new Socialist government PSOE, cancelled the Plan Hidrológico Nacional (National Hydrologic Plan), claiming that the ecological impact on the protected areas of the river had been underestimated, as well as the real cost of the project, having repercussions on the final cost of the transferred water (0.91 €/m^3 versus the 0.39 €/m^3 initial expectations) (PHN 2000). The new Plan AGUA (Actuaciones para la Gestión y la Utilización del Agua) (PHN 2005) was then introduced, based on supplementing existing supplies using desalinated water. More than 100 points of action such as the construction of large seawater RO desalination plants, water reuse, modernization and expansion of existing facilities among others were introduced (Molina and Casañas 2010). Nevertheless, the different policies between the national and local governments locked the completion and tender processes of the desalination plants, to the point that currently only 17 of the 51 planned

desalination plants are working, and 15 still under construction. However, due to the underestimation of the final cost of the desalted water (which will be two or three times the expected initial cost ~0.30€), and the change in the government at the end of 2011, after the right-wing party PP (critical with the use of desalination plants during the previous mandate) winning the elections, plants that are ready for the production of desalted water are working at the 17% of their full capacity. The possibility of returning to the original National Hydrologic Plan has been presented, but the European Union is pushing to increase the use of the desalination plants, after funding the project with approximately 1500 million of euros. In conclusion, all these aspects give Spain a great potential in seawater desalination, but due to the mentioned difficulties coupled with the current economic crisis, its future is uncertain.

3.5 **Challenges**

Although energy consumption of RO desalination plants has decreased significantly during the last decades, the total cost of the desalinated water production from seawater is not decreasing as much due to the increased cost of electricity, which usually represents 40-45% of the total cost of the produced water (Lenntech 2013). The need of cutting down costs is focusing the investigations in the field to increase the membrane efficiency and to optimize plant operation.

Modeling of RO processes is important to understand the behavior both in steady state and dynamic operation of water production plants. Specific models taking into account the characteristics of the RO plant and the equipment used are needed to optimize water production and design robust process control strategies (Jamal, Khan et al. 2004; Abbas 2006; Bartman, Christofides et al. 2009; Bartman, McFall et al. 2009). The development of first principle deterministic models of RO plants behavior requires fundamental knowledge of the complex physical phenomena that govern plant operation dynamics (i.e. interactions between membrane surface and solution, scaling, concentration polarization and fouling) (Belfort, Davis et al. 1994; Sablani, Goosen et al. 2001; Shirazi, Lin et al. 2010). For instance, fouling may occur during the operation of a RO plant decreasing its performance due to permeate flux decline. An increase in pressure is necessary to maintain the fresh

water production constant leading to higher energy consumption. Maintenance costs also increase due to the membrane cleaning or replacement that may be accelerated by fouling. Practical predictive fouling models that account for the interplay of various fouling mechanisms are not able to give accurate predictions or are not available for industrial size plants (Liu and Kim 2008; Gray, Semiat et al. 2011). In addition, factors such as temporal variability of feed composition, diurnal variations, and the inability of realistically quantify the real-time variability of feed fouling propensity need to be taken into account to develop realistic plant models.

Relative to deterministic models, data-driven algorithms (i.e., models based on plant data) provide a more effective way to predict RO plant behavior since they only require historic data instead of the specific process parameters needed by deterministic models (Richard Bowen, Jones et al. 2000), and might overcome the drawbacks that they present. Due to their ability to predict complex and non-linear systems accurately, these models are able to recognize and learn the characteristics of the plants (e.g., pressure drop, membrane permeability) that affect overall process performance, and can self-adapt to changes in operating conditions (Dirion, Cabassud et al. 1996).

Monitoring systems are necessary to control the plant and try to get the maximum performance of it, but there can be faults in the sensors that interfere the proper functioning of it. In order to avoid such situations, fault detection and data imputation systems are needed. The two main techniques to detect and isolated faults in the systems are Model-based and Signal-Processing techniques.

4. Self-Organizing Maps as a visualization tool and an exploratory data analysis

The Self-Organizing Map (Kohonen, 1990) is an unsupervised learning neural network architecture used for visualization, classification and data analysis of high-dimensional data. It has been applied in many fields, from engineering to social science and economics. In this section, an overview of the SOM algorithm is given along with an explanation of its visualization capabilities.

4.1 **The Self-Organizing Map**

The Self-Organizing Map algorithm is a nonlinear, ordered, smooth mapping of high-dimensional input data onto a regular low dimensional grid that preserves the relations between the input patterns (Kohonen 1997). The mapping is implemented by the SOM in a similar way that the Vector Quantization (Linde et al., 1980). SOM units are modified via an adaptation process (i.e., learning) which affects some predefined topological neighborhood producing both, an ordered and descriptive distribution of the original data and a vector quantization. Additionally, SOM can be considered as a nonlinear projection of the probability density function of the high-dimensional input data vector onto a low-dimensional (usually two-dimensional) map.

4.2 **Basics of the Self-Organizing Map algorithm**

The SOM algorithm is implemented in the following way. Assume that the set of input variables $\{\xi_j\}$ is a real vector $x=[\xi_1,\xi_2,...,\xi_n]^{\mathrm{T}} \in \Re^n$ and that each unit (i.e., neuron) in the SOM array has an associated parametric real vector $m_i=[\mu_1,\mu_2,...,\mu_{in}]^{\mathrm{T}} \in \Re^n$ which is called a model. Using a general distance measure between x and m_i defined as $d(x, m_i)$, the image of an input vector x on the SOM array is defined as its best matching (i.e., with a minimum distance) SOM unit m_c,

$$c=\arg\min_i \{d(x,m_i)\} \tag{8}$$

where *i* is the number of units of the SOM and *c* is the index of the best matching unit (i.e., closest SOM unit). The goal of the SOM is to create a map represented by the neurons $\boldsymbol{m_i}$ ordered in such a way that they are descriptive of the distribution of $\boldsymbol{x}$. In order to accomplish this, during the learning process (i.e., when the nonlinear projection is formed), the SOM units that are topographically close in the map up to a certain geometric distance activate each other to learn something from the same input $\boldsymbol{x}$. This results in a local smoothing effect on the weight vectors of the neurons in this neighborhood, which leads to global ordering when continued learning is carried out (Figure 3). The smoothing function $\boldsymbol{h_{ci}(t)}$ used during learning is the so-called neighborhood function. Neighborhood functions are centered in the best matching unit $\boldsymbol{c}$. One of the most widely applied neighborhood functions, is the Gaussian kernel,

$$h_{ci}(t) = \alpha(t) \cdot \exp\left(-\frac{\left\| r_c - r_i \right\|^2}{2 \cdot \sigma^2(t)} \right), \tag{9}$$

where $\boldsymbol{r_c}$ and $\boldsymbol{r_i}$ are the locations of units *c* and *i* on the map grid respectively, $\alpha(t)$ is the learning rate factor which takes values $0 < \boldsymbol{\alpha(t)} < 1$, and $\boldsymbol{\sigma(t)}$ is the width of the Gaussian. The update rule for the traditional SOM algorithm when the Euclidean distance is used as the measure of the distances is:

$$m_i(t+1) = m_i(t) + h_{ci}(t)\left[x(t) - m_i(t) \right] \tag{10}$$

The learning process is repeated until the SOM stabilizes (i.e., the SOM units are not modified anymore).

4.3 **SOM initialization and training**

The number of SOM units, map dimension, grid, lattice and shape need to be set during the SOM initialization. One needs to take in to account that the complexity of the algorithm increases exponentially with the number of map units, although the map has to be big enough to accommodate all training data. The most commonly used grid lattice is the hexagonal, since it has only 6 neighbors at the same distance per unit, while the rectangular lattice has 8. It is recommended that the shape of the map grid corresponds to the data manifold. Sheet map shapes are simple to apply, but present border effects, due to the presence of map units with reduced neighborhood. In order to avoid the border effects, cylindrical and toroidal map shapes can be applied. Typically, initialization can be carried out in three different ways: randomly, with random selected samples from the input data set or based on a Principal Component Analysis (PCA). In the latter case the initial codebook vectors lie in the two dimensional linear subspace spanned by the two eigenvectors which obtained largest eigenvalues in the autocorrelation matrix of $\boldsymbol{x}$.

The SOM is an unsupervised competitive data-driven learning approach where the map units compete to become specific detectors of certain data features. The map units contain *n*-dimensional vectors called *codebook,* where *n* is the same dimension of the input data $\boldsymbol{x}$. During the training, the similarities between $\boldsymbol{x}$ and the codebook vectors are calculated in order to find the *best matching unit* (BMU), using the following equation,

$$\left\| x - m_{bmu} \right\| = \min_i \left\{ \left\| x - m_i \right\| \right\} \tag{11}$$

Then the codebooks of the BMU and its topological neighborhood units are modified in order to increase the degree of matching with the input space. As it can be seen in Figure 3, the modifications of the SOM units always take place in such a direction that they match better with the input.

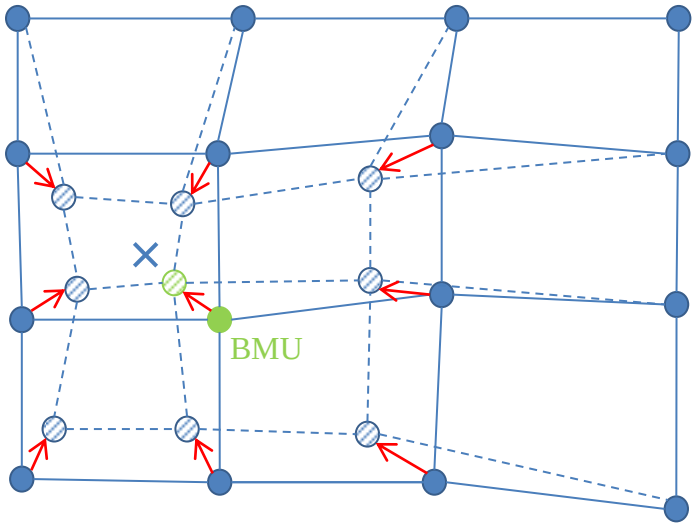

Figure 3. Updating the best matching unit (BMU) marked in green and its neighbors towards the input sample marked with an X. The solid and dashed lines correspond to the state of the map before and after updating respectively.

There are two phases that can be distinguished during the training which are controlled by the learning rate $\alpha(t)$ and the neighborhood radius $\sigma(t)$. Both $\alpha(t)$ and $\sigma(t)$ are monotonically decreasing functions of time, which start with high values at the beginning of the training, where a rough tuning is achieved (ordering phase), and get smaller values at the end of the training, where a fine tuning is attained (convergence phase).

4.4 **SOM Quality Measures**

The most commonly applied techniques to measure the quality of a SOM's projection are, Quantization Error (QE), Topgraphic Error (TE) and Average Distortion Measure (ADM). Their two main objectives are to measure the quality of the continuity of mapping and the topology preservation, and to measure the mapping resolution.

The Quantization Error measures the average distance between the input patterns $\boldsymbol{x_i}$ and the reference vector $\boldsymbol{m_c}$ associated to their Best Matching Units (Kohonen 1997).

$$QE = \frac{\sum_{i=1}^{n} d\left(x_i, m_c\right)}{n} \tag{12}$$

Where ***n*** is the total number of input patterns. Low QE values mean a good mapping of the input data, but in some cases may also be associated with an over fitted model.

The Topographic Error (TE) represents the proportion of the total number sample vectors ***n*** for which the two best matching units are not adjacent, which indicates a local discontinuity of the mapping (Kiviluoto 1996),

$$TE = \frac{1}{n}\sum_{i=1}^{n} u(x_i) \quad , \tag{13}$$

where *u(xi)* =0 if the first and second *BMUs* are adjacent, and one otherwise. *TE* only indicates the portion of the local neighborhoods correctly mapped, but it does not describe the kind of incorrect mapping (i.e., it does not indicate how far the two BMUs are). Consequently, when high *TE* values are obtained they indicate that there is a high portion of neighborhoods with topographic errors.

In the case of a SOM of discrete data and a fixed neighborhood function, the SOM quality can be measured by a local energy function which can calculate an error value for the whole map.

$$TE = \sum_{k}\sum_{i} h_{ci} \left\| x_k - m_i \right\|^2 \tag{14}$$

This energy function corresponds to the k-means vector quantization and takes into account the distance of each input vector to all the reference vectors weighted by the neighborhood kernel.

4.5 **SOM visualization**

Most of the SOM-based techniques that can be used to visualize relationships among data once the reference vectors have been ordered in the map grid represent data in two dimensions. Some of the most common SOM visualization techniques are explained below.

Component planes

Each component plane (or *c-plane*) shows the relative values of each one of the components of the codebook vectors by plotting them with a color code or a gray scale over the SOM grid. The number of component planes of a SOM is equal to the size of the input data dimension. Relationships between different component planes (i.e., input variables) can be inferred by observing simultaneously at several component planes.

Unified Distance Matrix

The Unified Distance Matrix (or U-Matrix) shows the relative distances between adjacent codebook vectors (or units). For a two-dimensional grid (a*b)-extension, the u-matrix has ((a-1)*(b-1))-extension. The average distance uhi of a codebook vector wi to its immediate neighbors is calculated using:

$$uh_i = \frac{1}{n}\sum_j d\left(w_i, w_j\right),\ j \in N(i),\ n = \left|N(i)\right|, \tag{15}$$

where $d(w_i,w_j)$ represents the distance between the vectors *i* and *j* and $N(i)$ are the neighbor units of vector *i*. Usually the U-matrix is represented as a 2D gray scale where light colors correspond to low distance values while darker colors represent high ones. The U-Matrix is very useful to detect clusters, represented by light areas, and the boundaries between them represented by dark gray regions. Nevertheless it can be difficult to detect visually the different clusters. Clustering algorithms such as K-means are able to detect and cluster units of similar structural characteristics. The clustering quality is evaluated by means of the Davies-Bouldin index (Davies and Bouldin 1979), which takes into account both the error caused by representing the data vectors with their cluster centroids (intra cluster diversity), and the distance between clusters (inter-cluster distance). Clustering parameters and the optimal number of clusters are optimized based on the minimization of the Davies-Bouldin index.

P-Matrix

The p-matrix calculates the data density around a codebook vector and represents the number of input data points within a hypersphere with a Pareto radius around each weigh vector (Fincke, Lobo et al. 2008). The values contained in the cells of the p-matrix are the p-heights *ph(i)*, which are calculated by,

$$ph_i = \left| \left\{ x \in E \mid d\left(x, w_i\right) < r,\ r \in \mathbb{R}^+ \right\} \right| \tag{16}$$

where $d(x,w_i)$ represents the distance between the codebook w_i and input vector x, E is the input data set and r is the Pareto radius around w_i. SOM units with high p-height values are located in dense areas of the data space, while the ones with low p-height values are located in sparsely populated areas. P-matrix may look similar to hit maps, but the main difference between them is that in the p-matrix an input vector can be part of more than one unit, while in the hit map each input data vector is assigned only to one map unit.

5. Support Vector Machines

Support Vector Machine (SVM) is a supervised learning algorithm developed by Vladimir Vapnik that generates input-output mapping functions from a set of labeled training data. SVM can be applied either for classification (i.e., to categorize the input data) or for regression. For classification, input data is transformed to a high-dimensional feature space by using nonlinear kernel functions. Maximum-margin hyperplanes based only on a subset of training data near the boundaries are then created in the high-dimensional feature space, where the transformed data is more separable than in the original input space. For regression the model produced by Support Vector Regression ignores any training data sufficiently close to the model prediction.

5.1 **Basics of the Support Vector Machines**

SVM are applied when there is an unknown nonlinear dependency $y = f(x)$ between a high dimensional input vector ***x*** and an output ***y*** (target), which can be a scalar or a vector in the case of a multiclass SVM. The only available information about the function is a training data set $D = \{(x_i, y_i) \in X \times Y\}$, $i = 1, l$, where l stands for the number of training data pairs (i.e., equal to the size of the training data set *D*). Unlike in classic statistical inference, SVM parameters do not need to be defined prior to the training, since they are data-driven to match the model capacity to data complexity. In order to develop models with a good generalization property, the constructive approach followed by the SVM consists on keeping the value of the training error (i.e., approximation error or empirical risk) fixed (to some acceptable level), and minimize the confidence interval. This cost function utilized for the development of SVM models is expressed as:

$$R = \sum_{i=1}^{l} L_{\varepsilon} + \Omega(l, h) \ , \tag{17}$$

where $L\varepsilon$ is a loss function, h is a Vapnik-Chervonenkis (VC) dimension and Ω is a function bounding the capacity of the learning machine. For classification problems, $L\varepsilon$ is a 0-1 loss function, while in regression applications, $L\varepsilon$ is the Vapnik's ε–intensitivity loss (error) function

$$L_{\varepsilon} = |y - f(x,w)|_{\varepsilon} = \begin{cases} 0, & \text{if } |y - f(x,w)| \le \varepsilon \\ |y - f(x,w)| - \varepsilon, & \text{otherwise.} \end{cases} \tag{18}$$

where ε is the radius of a tube within which the regression function must lie after the learning and *w* is the set of parameters also known as weights. SVM generates a model with minimized VC dimension therefore the expected probability error of the model is also low, which leads to good generalization (i.e., low errors when predicting previously unseen data).

In the simplest pattern recognition cases, SVM create a linear separating hyperplane to classify the different patterns with a maximal margin. The hyperplane is obtained as the

result of a constrained nonlinear optimization problem, which has a quadratic cost function and linear constrains.

When the classes are not linearly separable in the original input space, the input space is non-linearly transformed into a higher dimensional feature space. This transformation is usually carried out by polynomial, sigmoidal, RBF mappings (such as Gaussians), or multiquadratics or different spline functions. Thanks to the nonlinear transformation, nonlinear separable problems in the original input space become linearly separable in the feature space. After the transformation is done, the SVM creates the linear optimal separating hyperplane in this feature space, in the same way as for linearly separable problems in the input space. When SVM are applied for regression, the training phase consist on finding the relationship $f_a(x,w)$ (i.e., approximation function) between the input space *X* and the output space *Y*, using data *D*. In classification applications, the function obtained during the training is the decision boundary (i.e., separating function) that separates the data within the input space. The function $f_a(x,w)$ which belongs to the hypothesis space of functions $H(f_a \in H)$ is selected as the one that minimizes the risk function $R(w)$.

The main problem found during the training is that the number of solutions of the learning problem is infinite. In other words, in the case of regression (**Fig. 4**), the number of functions that interpolate data performing training errors equal to zero is infinite, while in the case of classification, there is an infinite number of functions that perfectly separate the training patterns.

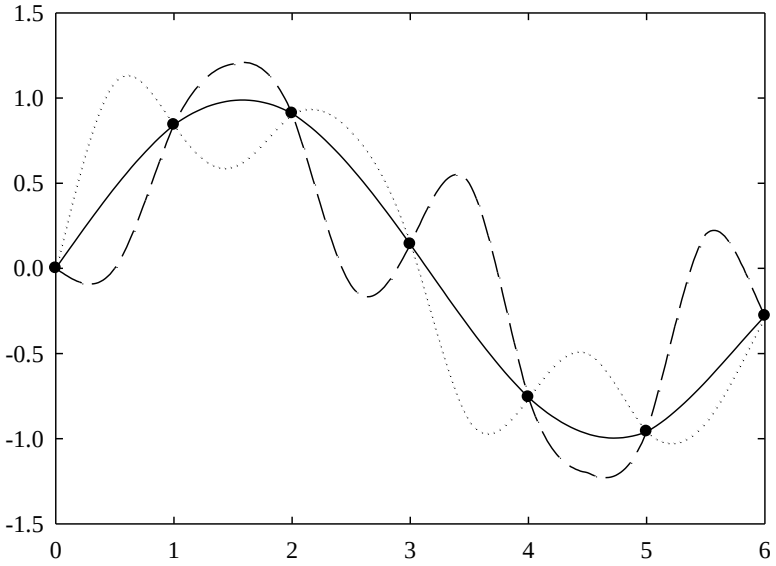

Figure 4. Two out of the infinite interpolating functions which perform a training error equal to 0. The dashed and dotted lines correspond to bad (overfitted) models of the real function $y = \sin(x)$ (solid line).

The three functions plotted in **Figure 4** perform a training error equal to zero. Note that the training error can be minimized during the training. Nevertheless, only the function represented by the solid line is able to generalize and model the true dependency between ***x*** and ***y***. The generalization capability of the interpolation functions is not always controlled during the training, since it depends on the specific training data pairs used. In order to solve this problem and find a good generalizing function, the hypothesis space *H* of the approximating functions is restricted to a smaller set of functions while the complexity (or flexibility) of the approximating functions is controlled (Vapnik 1982).

5.2 **Support vector machines for regression**

Support Vector techniques were initially developed for classification problems, although later they were modified to be applied in regression (Drucker, Burges et al. 1997; Vapnik, Golowich et al. 1997).The objective of the Support Vector Regression is to estimate the dependence of the output variable $y \in \Re$ on an n-dimensional input variable ***x*** (Smola and Schölkopf 2004). It consists in learning the input-output relationship (dependency) $f(x)$

from the training data given to the learning machine. The training data set $D = \left\{ \left[x(i), y(i) \right] \in \Re^n \times \Re,\ i = 1, ..., l \right\}$ consist of l pairs $\left(x_1, y_1 \right)$, $\left(x_2, y_2 \right), \ldots, \left(x_l, y_l \right)$. The inputs *x* are n-dimensional vectors $x \in \Re^n$ and $y \in \Re$ are continuous system responses.

5.2.1 **Application to linear regression problems**

The learning function developed by the SVM for regression is a linear regression hyperplane expressed as

$$f(x, w) = w^T x + b. \tag{19}$$

The error of approximation of the learning function is measured introducing the concept of the ε-insensitivity zone (**Fig. 5**), defined as the space included inside a tube of radius ε. Vapnik's ε-insensitivity loss (error) function defined as:

$$E\left(x, y, f \right) = \left| y - f\left(x, w \right) \right|_\varepsilon = \begin{cases} 0 & \text{if } \left| y - f\left(x, w \right) \right| \le \varepsilon, \\ \left| y - f\left(x, w \right) \right| - \varepsilon, & \text{otherwise.} \end{cases} \tag{20}$$

or as,

$$e\left(x, y, f \right) = \max\left(0, \left| y - f\left(x, w \right) \right| - \varepsilon \right). \tag{21}$$

Accordingly, the loss function equals 0 when the difference between the measured y_i and the predicted value $f\left(x_i, w \right)$ is less than ε (i.e., the predicted value is within the tube). By contrast, when the predicted values are outside the tube, the loss function is equal to the magnitude of the difference between the predicted value and the radius ε of the tube.

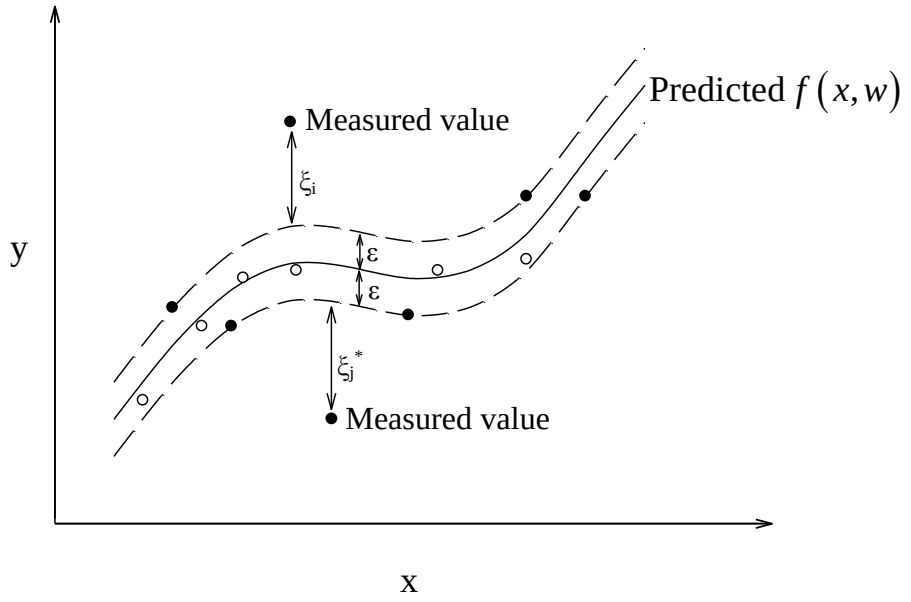

Figure 5. Representation of a one-dimensional support vector regression estimation function $f(x,w)$. The filled dots • in the boundary or outside the tube of radius ε are the support vectors, while the empty dots ∘ inside are not and ξ_i and ξ_i^* are slack variables for the measured values above and below the ε-tube respectively.

Firstly, the formulation of the SVR algorithm for linear problems is presented below. The empirical error term R_{emp} from equations (20 and 21) is measured simultaneously to the minimization of the confidence term Ω by minimizing $w^T w$.

$$R_{emp}^{\varepsilon}(w,b) = \frac{1}{l}\sum_{i=1}^{l}\left|y_i - w^T x_i - b\right|_{\varepsilon} \tag{22}$$

Since the objective is to minimize simultaneously the empirical risk R_{emp}^{ε} and $\|w\|^2$, a linear regression hyperplane $f(x,w) = w^T x + b$ is constructed by minimizing

$$R = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\left|y_i - f(x_i,w)\right|\varepsilon. \tag{23}$$

From equation (20) and figure 5, it is deduced that for the training data outside the ε–tube,

$\left|y - f(x,w)\right| - \varepsilon = \xi$ for data "above" the ε –tube, or

$$\left|y - f\left(x, w\right)\right| - \varepsilon = \xi^* \quad \text{for data "below" the } \varepsilon \text{–tube.}$$

Therefore, the minimization of the risk R can be expressed as

$$R_{w,\xi,\xi^*} = \left[\frac{1}{2}\|w\|^2 + C\left(\sum_{i=1}^{l}\xi_i + \sum_{i=1}^{l}\xi_i^*\right)\right], \tag{24}$$

under the constrains

$$y_i - w^T x_i - b \le \varepsilon + \xi_i, \quad i = 1, l, \tag{25}$$

$$w^T x_i + b - y_i \le \varepsilon + \xi_i^*, \quad i = 1, l, \tag{26}$$

$$\xi_i \ge 0, \quad \xi_i^* \ge 0, \qquad i = 1, l. \tag{27}$$

where the constant C is chosen by the user and controls the trade-off between the approximation error and the weight vector norm $\|w\|$. The choice of a high value for C implies a penalty for large errors (i.e., forces the slack variables ξ_i and ξ_i^* to be small), which leads to an approximation error decrease, achieved by increasing the weight vector norm $\|w\|$. At the same time, a large value of the weight vector $\|w\|$ increases the confidence term Ω, which does not guarantee a small generalization of the approximation function. The radius ε of the ε –tube is also chosen by the user. The solution of the constrained optimization problem in (24) is carried out by forming a primal variables Lagrangian,

$$\begin{aligned} L_p\left(w, b, \xi_i, \xi_i^*, \alpha_i, \alpha_i^*, \beta_i, \beta_i^*\right) = {} & \frac{1}{2}w^T w + C\sum_{i=1}^{l}\left(\xi_i + \xi_i^*\right) - \sum_{i=1}^{l}\left(\beta_i^*\xi_i^* + \beta_i\xi_i\right) \\ & - \sum_{i=1}^{l}\alpha_i\left[w^T x_i + b - y_i + \xi + \xi_i\right] \\ & - \sum_{i=1}^{l}\alpha_i^*\left[y_i - w^T x_i - b + \xi + \xi_i^*\right] \end{aligned} \tag{28}$$

where α_i, α_i^*, β_i and β_i^* are non-negative Lagrange multipliers. The primal variables Lagrangian $L_p\left(w, b, \xi_i, \xi_i^*, \alpha_i, \alpha_i^*, \beta_i, \beta_i^*\right)$ has to be minimized with respect to w, b, ξ_i, and

ξ_i^* and maximized with respect to α_i, α_i^*, β_i and β_i^*. Therefore, the function has a saddle point at the optimal solution $\left(w_0, b_0, \xi_{i0}, \xi_{i0}^*\right)$, where the partial derivatives of L_p are zero:

$$\frac{\partial L_p\left(w_0, b_0, \xi_{i0}, \xi_{i0}^*, \alpha_i, \alpha_i^*, \beta_i, \beta_i^*\right)}{\partial w} = w_0 - \sum_{i=1}^{l}\left(\alpha_i - \alpha_i^*\right)x_i = 0 \ , \tag{29}$$

$$\frac{\partial L_p\left(w_0, b_0, \xi_{i0}, \xi_{i0}^*, \alpha_i, \alpha_i^*, \beta_i, \beta_i^*\right)}{\partial b} = \sum_{i=1}^{l}\left(\alpha_i - \alpha_i^*\right) = 0 \ , \tag{30}$$

$$\frac{\partial L_p\left(w_0, b_0, \xi_{i0}, \xi_{i0}^*, \alpha_i, \alpha_i^*, \beta_i, \beta_i^*\right)}{\partial \xi_i} = C - \alpha_i - \beta_i = 0 \ , \tag{31}$$

$$\frac{\partial L_p\left(w_0, b_0, \xi_{i0}, \xi_{i0}^*, \alpha_i, \alpha_i^*, \beta_i, \beta_i^*\right)}{\partial \xi_i^*} = C - \alpha_i^* - \beta_i^* = 0 \, . \tag{32}$$

Substituting (29), (30), (31) and (32) into (28) yields the dual optimization problem,

$$\begin{aligned} L_p\left(\alpha_i, \alpha_i^*\right) &= -\frac{1}{2}\sum_{i,j=1}^{l}\left(\alpha_i - \alpha_i^*\right)\left(\alpha_j - \alpha_j^*\right)x_i^T x_j - \varepsilon\sum_{i=1}^{l}\left(\alpha_i + \alpha_i^*\right) + \sum_{i=1}^{l}\left(\alpha_i - \alpha_i^*\right)y_i \\ &= -\frac{1}{2}\sum_{i,j=1}^{l}\left(\alpha_i - \alpha_i^*\right)\left(\alpha_j - \alpha_j^*\right)x_i^T x_j - \sum_{i=1}^{l}\left(\varepsilon - y_i\right)\alpha_i - \sum_{i=1}^{l}\left(\varepsilon + y_i\right)\alpha_i^* \end{aligned} \tag{33}$$

subject to

$$\sum_{i=1}^{l}\alpha_i^* = \sum_{i=1}^{l}\alpha_i \ \ or \ \ \sum_{i=1}^{l}\left(\alpha_i - \alpha_i^*\right) = 0 \tag{34}$$

$$0 \le \alpha_i \le C \quad i = 1, l \ , \tag{35}$$

$$0 \le \alpha_i^* \le C \quad i = 1, l \ . \tag{36}$$

This standard quadratic optimization problem can also be expressed in a matrix notation:

$$\text{minimize} L_d\left(\alpha\right) = -0.5\alpha^T \mathrm{H}\alpha + \mathrm{f}^T\alpha \, , \tag{37}$$

subject to (34-36), where $\alpha = \left[\alpha_1, \alpha_2, ... \alpha_l, \alpha_1^*, \alpha_2^*, ... \alpha_l^* \right]^T$, H=$\left[\text{G -G; -G G} \right]$, G is an (*l*,*l*) matrix containing $G_{ij} = \left[x_i^T x_j \right]$ and $\text{f} = \left[\varepsilon - y_1, \varepsilon - y_2, ..., \varepsilon - y_l, \varepsilon + y_1, \varepsilon + y_2, ..., \varepsilon - y_l \right]^T$.

The result of the learning stage are *l* Langrange multiplier pairs $\left(\alpha_i, \alpha_i^* \right)$. The number of support vectors (SVs) does not depend on the dimensionality of the input space data, but is equal to the number of nonzero parameters α_i or α_i^*. Additionally, the optimal solution must accomplish the following conditions:

$$\alpha_i \left(w^T x_i + b - y_i + \varepsilon + \xi_i \right) = 0, \tag{38}$$

$$\alpha_i^* \left(-w^T x_i - b + y_i + \varepsilon + \xi_i^* \right) = 0, \tag{39}$$

$$\beta_i \xi_i = \left(C - \alpha_i \right) \xi_i = 0 , \tag{40}$$

$$\beta_i^* \xi_i^* = \left(C - \alpha_i^* \right) \xi_i^* = 0 . \tag{41}$$

From (39) it can be stated than for $0 < \alpha_i < C$, $\xi_i = 0$, and similarly from (41) for $0 < \alpha_i^* < C$, $\xi_i^* = 0$. Consequently, for $0 < \alpha_i$, $\alpha_i^* < C$ from (38) and (39) become:

$$w^T x_i + b - y_i + \varepsilon = 0, \tag{42}$$

$$-w^T x_i - b + y_i + \varepsilon = 0. \tag{43}$$

Thereby, the value of α_i for all the data points satisfying $y - f\left(x \right) = +\varepsilon$ must be $0 < \alpha_i < C$. In the same way α_i^* for the data points fulfilling $y - f\left(x \right) = -\varepsilon$ must be $0 < \alpha_i^* < C$. The data points accomplishing this conditions are the ones called free support vectors. The bias term *b* can be then computed as follows:

$$b = y_i - w^T x_i - \varepsilon, \text{ for } 0 < \alpha_i < C, \tag{44}$$

$$b = y_i - w^T x_i + \varepsilon, \text{ for } 0 < \alpha_i^* < C. \tag{45}$$

From (39) and (40), it can also be observed that when $\xi_i > 0$ and $\xi_i^* > 0$ (i.e., the data points are outside the ε–tube), both α_i and α_i^* are equal to C. The points corresponding to these data are so-called bounded support vectors. By contrast, α_i and α_i^* for all the data points satisfying $|y - f(x)| < \varepsilon$ (i.e., the data points within the ε–tube) are equal to zero, in other words, they do not construct the function $f(x)$ (i.e., they are not support vectors).

When the Langrange multipliers are calculated by using (29), the optimal weight vector of the regression hyperplane can be obtained by

$$w_0 = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*) x_i , \tag{46}$$

and then the optimal regression hyperplane can be obtained by

$$f(x, w) = w_0^T x + b = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*) x_i^T x + b. \tag{47}$$

5.2.2 **Application to nonlinear regression problem**

The generalization for nonlinear regression problems is developed by transforming the mapping in the original input space to a mapping in the *feature space*. This can be done by using kernel functions which produces very high dimensional (sometimes infinite) mappings. Consequently, the nonlinear regression function in the input space is produced from a linear regression hyperplane in the *feature space*.

The input vectors $x \in \Re^n$ are mapped into vectors $\Phi(x)$ in a higher dimensional *feature space F* (where Φ represents mapping $\Re^n \to \Re^f$), and then the linear regression is carried out in this feature space. The input space (x-space) is spanned by components x_i of an input vector x and the feature space F (Φ-space) is spanned by components $\phi_i(x)$ of a vector $\Phi(x)$. The objective of this mapping is to make the learning algorithm able to develop a linear regression hyperplane in the feature space. The solution of the regression hyperplane

$f = w^T \Phi(x) + b$ (which is linear in the feature space *F*) creates a nonlinear regressing hypersurface in the original input space.

The learning problem can be formulated as the maximization of ta dual Lagrangian (37) with the Hessian matrix structured in the same way as in the linear case (i.e., $\mathrm{H}=[\mathrm{G}\ -\mathrm{G};-\mathrm{G}\ \mathrm{G}]$), but with the changed Grammian (kernel) matrix **G**:

$$\mathrm{G} = \begin{bmatrix} \mathrm{G}_{11} \cdots \mathrm{G}_{11} \\ \vdots \ \mathrm{G}_{ii} \ \vdots \\ \mathrm{G}_{l1} \cdots \mathrm{G}_{ll} \end{bmatrix}, \tag{48}$$

where $\mathrm{G}_{ij} = \Phi^T(x_i)\Phi(x_j) = K(x_i, x_j),\ i, j = 1, l.$

Once α_i and α_i^* are calculated, the optimal weighting vector of the kernels expansion can be obtained as:

$$\mathrm{v}_0 = \alpha - \alpha^*. \tag{49}$$

Accordingly, the best nonlinear regression function obtained from the optimal weighting vector $\mathbf{v_0}$ and the kernel matrix **G** is:

$$f(x, w) = \mathrm{G}\,\mathrm{v}_0 + b \tag{50}$$

Equation (50) results from replacing the term x_i by $\Phi(x_i)$ in all the equations from (25) to (47), which implies that (46) and (47) now become (51) and (52) respectively:

$$w_0 = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*)\Phi(x_i), \tag{51}$$

$$\begin{aligned} f(x, w) = w_0^T \Phi(x) + b &= \sum_{i=1}^{l} (\alpha_i - \alpha_i^*)\Phi^T(x_i)\Phi(x) + b\ , \\ &= \sum_{i=1}^{l} (\alpha_i - \alpha_i^*) K(x_i, x) + b\ . \end{aligned} \tag{52}$$

When the bias *b* is used explicitly, likewise in (50), it can be calculated from the upper support vectors as,

$$\begin{aligned} b &= y_i - \sum_{j=1}^{Nu} \left(\alpha_j - \alpha_j^*\right) \Phi^T\left(x_j\right) \Phi\left(x_i\right) - \varepsilon \\ &= y_i - \sum_{j=1}^{Nu} \left(\alpha_j - \alpha_j^*\right) K\left(x_i, x\right) - \varepsilon, \qquad \text{for } 0 < \alpha_i < C, \end{aligned} \tag{53}$$

and from the lower support vectors as,

$$\begin{aligned} b &= y_i - \sum_{j=1}^{Nl} \left(\alpha_j - \alpha_j^*\right) \Phi^T\left(x_j\right) \Phi\left(x_i\right) + \varepsilon \\ &= y_i - \sum_{j=1}^{Nl} \left(\alpha_j - \alpha_j^*\right) K\left(x_i, x\right) + \varepsilon, \qquad \text{for } 0 < \alpha_i^* < C, \end{aligned} \tag{54}$$

where *Nu* and *Nl* are the number of free upper and lower vectors respectively. Note than in (53) $\alpha_j^* = 0$ and so is $\alpha_j = 0$ in (54).

The three most important parameters that the user needs to set for the construction of the Support Vector Regression are the insensitivity zone ε, the penalty parameter C (which controls the trade-off between the training error and the model complexity), and the parameters of the kernel chosen (variances of a Gaussian, order of the polynomial, shape of the inverse multiquadratics). In order to obtain the optimal values for the SVR parameters, a cross-validation is carried out. In cases where the data is not too noisy (mainly without huge outliers), it is possible to solve the problem by setting the penalty parameter C to infinity and control the modeling by tuning the insensitivity zone ε and kernel shape parameters only.

Although SVM work perfectly for not too large data basis, when the number of data points is large (l>2,000), the QP problems becomes very memory consuming, which makes it difficult to handle even for the current standard computers. To address the problem of dealing with large data sets, three different approaches have been developed, which basically consist in decomposing the data (Vapnik 1995; Osuna, Freund et al. 1997) or carry out the optimization sequentially (Platt 1999).

6. References

Abbas, A. (2006). "Model predictive control of a reverse osmosis desalination unit." Desalination 194(1–3): 268-280.

Bartman, A. R., P. D. Christofides, et al. (2009). "Nonlinear Model-Based Control of an Experimental Reverse-Osmosis Water Desalination System." Industrial & Engineering Chemistry Research 48(13): 6126-6136.

Bartman, A. R., C. W. McFall, et al. (2009). "Model-predictive control of feed flow reversal in a reverse osmosis desalination process." Journal of Process Control 19(3): 433-442.

Belfort, G., R. H. Davis, et al. (1994). "The behavior of suspensions and macromolecular solutions in crossflow microfiltration." Journal of Membrane Science 96(1–2): 1-58.

Davies, D. L. and D. W. Bouldin (1979). "A Cluster Separation Measure." Pattern Analysis and Machine Intelligence, IEEE Transactions on PAMI-1(2): 224-227.

Dirion, J. L., M. Cabassud, et al. (1996). "Development of adaptive neural networks for flexible control of batch processes." The Chemical Engineering Journal and the Biochemical Engineering Journal 63(2): 65-77.

Drucker, H., C. J. C. Burges, et al. (1997). Support vector regression machines. Advances in Neural Information Processing Systems 9, MIT Press, Cambridge, MA, USA.

Fincke, T., V. Lobo, et al. (2008). "Visualizing self-organizing maps with GIS." GI Days.

Gleick, P. H. (2011). The World's Water, Volume 7. The Biennial Report on Freshwater Resources. Washington D.C., Island Press.

Gray, S., R. Semiat, et al. (2011). 4.04 - Seawater Use and Desalination Technology. Treatise on Water Science. W. Editor-in-Chief: Peter. Oxford, Elsevier: 73-109.

Jamal, K., M. A. Khan, et al. (2004). "Mathematical modeling of reverse osmosis systems." Desalination 160(1): 29-42.

Johnson, J. and M. Busch (2010). "Engineering Aspects of Reverse Osmosis Module Design." Desalination and Water Treatment 15(1-3): 236-248.

Kiviluoto, K. (1996). Topology preservation in self-organizing maps. Neural Networks, 1996., IEEE International Conference on.

Kohonen, T. (1997). Self-organizing maps, Springer-Verlag Berlin Heidelberg New York, Inc.

Kucera, J. (1997). "Properly Apply Reverse Osmosis." Chemical engineering Progress: 54-61.

Lenntech. (2013). "http://www.lenntech.com/." 2013.

Liu, Q.-F. and S.-H. Kim (2008). "Evaluation of membrane fouling models based on bench-scale experiments: A comparison between constant flowrate blocking laws and artificial neural network (ANNs) model." Journal of Membrane Science 310(1–2): 393-401.

Molina, V. G. and A. Casañas (2010). "Reverse osmosis, a key technology in combating water scarcity in Spain." Desalination 250(3): 950-955.

Osuna, E., R. Freund, et al. (1997). Training support vector machines: an application to face detection. Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on, IEEE.

PHN (2000). Plan Hidrológico Nacional: Análisis económicos. Madrid, Ministerio de Medio Ambiente: 390.

PHN (2001). Ley 10/2001, de 5 de julio, del Plan Hidrológico Nacional. Boletín Oficial del Estado. 161: 24228-24250.

PHN (2005). Ley 11/2005, de 22 de junio, por la que se modifica la Ley 10/2001, de 5 de julio, del Plan Hidrológico Nacional. 149: 21846-21856.

Platt, J. C. (1999). Fast training of support vector machines using sequential minimal optimization. Advances in kernel methods, MIT Press: 185-208.

Richard Bowen, W., M. G. Jones, et al. (2000). "Predicting salt rejections at nanofiltration membranes using artificial neural networks." Desalination 129(2): 147-162.

Sablani, S. S., M. F. A. Goosen, et al. (2001). "Concentration polarization in ultrafiltration and reverse osmosis: a critical review." Desalination 141(3): 269-289.

Shirazi, S., C.-J. Lin, et al. (2010). "Inorganic fouling of pressure-driven membrane processes — A critical review." Desalination 250(1): 236-248.

Smola, A. J. and B. Schölkopf (2004). "A tutorial on support vector regression." Statistics and Computing 14(3): 199-222.

Vapnik, V. (1982). Estimation of Dependences Based on Empirical Data: Springer Series in Statistics (Springer Series in Statistics), Springer-Verlag New York, Inc.

Vapnik, V., S. E. Golowich, et al. (1997). Support Vector Method for Function Approximation, Regression Estimation and Signal Processing. Advances in Neural Information Processing Systems 9 --- Proceedings of the 1996 Neural Information Processing Systems Conference (NIPS 1996), Dever, CO, USA, MIT Press, Cambridge, MA, USA.

Vapnik, V. N. (1995). The nature of statistical learning theory, Springer-Verlag New York, Inc.

7. Article 1. Data-driven models of steady state and transient operations of spiral-wound RO plant

Article 1

Authors: X. Pascual, H. Gu, A. Bartman, A. Zhu, A. Rahardianto, J. Giralt, R. Rallo, P. D. Christofides, Y. Cohen.

Title: Data-driven models of steady state and transient operations of spiral-wound RO plant.

Journal: Desalination, ISSN: 0011-9164

Volume: 316 **Pages:** 154-161 **Year:** 2013

ISI category: Chemical Engineering **Impact factor:** 3.041

Position in the category: 13/133

Cites: -

Data-driven models of steady state and transient operations of spiral-wound RO plant

Xavier Pascual [a], Han Gu [b], Alex Bartman [b], Aihua Zhu [b], Anditya Rahardianto [b],
Jaume Giralt [a], Robert Rallo [c], Panagiotis D. Christofides [b], Yoram Cohen [b, *]

[a] SCITA, Departament d'Enginyeria Química, Universitat Rovira i Virgili, Av. Països Catalans 26, 43007 Tarragona, Catalunya, Spain

[b] Department of Chemical and Biomolecular Engineering, Water Technology Research Center, University of California, 5531 Boelter Hall, Los Angeles, CA 90095-1592, USA

[c] BioCENIT, Departament d'Enginyeria Informàtica i Matemàtiques, Universitat Rovira i Virgili, Av. Països Catalans 26, 43007 Tarragona, Catalunya, Spain

* Corresponding author; Email: yoram@ucla.edu; Tel: 310-825-8766, Fax: 310-206-4107.

Abstract

The development of data-driven RO plant performance models was demonstrated using the support vector regression model building approach. Models of both steady state and unsteady state plant operation were developed based on a wide range of operational data obtained from a fully automated small spiral-wound RO pilot. Single output variable steady state plant models for flow rates and conductivities of the permeate and retentate streams were of high accuracy, with average absolute relative errors (AARE) of 0.70%-2.46%. Performance of a composite support vector regression (SVR) based model (for both streams) for flow rates and conductivities was of comparable accuracy to the single output variable models (AARE of 0.71%-2.54%). The temporal change in conductivity, as a result of transient system operation (induced by perturbation of either system pressure or flow rate) was described by SVR model, which utilizes a time forecasting approach, with performance level of less than 1% AARE for forecasting periods of 2 s to 3.5 min. The high level of performance obtained with the present modeling approach suggests that short-term performance forecasting models that are based on plant data, could be useful for advanced RO plant control algorithms, fault tolerant control and process optimization.

Keywords: *Desalination; data-driven models; Support vector regression, spiral-wound RO plant, process models*

1. Introduction

Water desalination by reverse osmosis (RO) membrane technology has been increasingly deployed for potable water production from seawater and water reuse application including municipal wastewater and agricultural drainage (AD) water. Most RO plants are designed to operate at relatively steady state conditions with traditional control strategies to attain the target permeate productivity and quality. Given the complexity of RO plants, plant process models, which consider specific plant characteristics and equipment, are needed to describe both steady state and dynamic plant operations in order to optimize water production and design robust process control strategies [1-4].

The development of first principle deterministic models of RO plant behavior requires fundamental knowledge of the complex physical phenomena that govern plant operational dynamics including, but not limited to, behavior of sensors and actuators, concentration polarization [5], membrane fouling [6] and mineral scaling [7]. For example, membrane fouling can lead to permeate flux decline when operating under constant transmembrane pressure or increased net driving pressure under constant flux operation [7]. Deterministic plant models that are a priori predictive of fouling and mineral scaling would clearly be of practical value; however, given the challenge of accounting for the complex interplay among various fouling [8] and scaling mechanisms [9], such models are lacking for industrial scale plants [8]. Admittedly, commercial RO system design software (e.g., Winflows [10], CAROL [11] and ROSA [12]), which are built on the basis of deterministic and semi-empirical models, can be used to simulate steady state operation of RO plants. Mechanistic computational (CFD) models of RO desalination have also been advanced over the last few decades [13] focusing on either simple membrane channel geometries or modeling of single membrane modules. Efforts to incorporate the impact of fouling on the operation of spiral-wound membranes in theoretical and CFD models have also been recently proposed [13-15] and hold promise for adoption in full-scale plant models. The use of both computational CFD models and software design packages for accurate simulation of real-time plant performance is limited since such models typically do not account for complex plant hydraulics, the evolution of fouling and mineral scaling throughout the plant, and plant equipment performance over time (e.g., pumps, valves, sensors, etc.).

Data-driven algorithms (i.e., models based on plant data) represent another class of models capable of providing an effective way of describing plant behavior making use of historical plant data without having to rely on predetermined process parameters that are needed by deterministic models [16]. Data-driven models are well suited for accurately describing complex and non-linear systems. Therefore, models can be trained to recognize and learn the characteristics of the plant that affect overall process performance. One advantage of data-driven models is that they can self-adapt (through incremental learning) to changes in operating conditions [17]. Data-driven models (e.g., based on process operational data) can be integrated within control systems [18] by facilitating the development of virtual sensors capable of inferring the properties of manufactured products [19], that provide the basis to improve plant control strategies [20] in addition to data-driven models of membrane based separation processes [21].

Over the past two decades, there has been a growing interest in developing data-driven models, based on machine learning methods, to describe membrane performances (e.g., transmembrane flux and rejection) and fouling in membrane separation processes that include microfiltration (MF), ultrafiltration (UF), nanofiltration (NF) and reverse osmosis (RO). Artificial neural networks (ANN) derived models for NF membrane salt rejection [22] were reported with high level of performance with average absolute deviation not greater than 5%. ANN based models of fouling of hollow fiber membranes were reported for a bench-scale system [8] enabling a single composite model for transmembrane pressure covering the various stages of fouling, while piecewise fitting was required when using deterministic fouling models (cake formation, surface blocking and pore blocking models). ANN based models [23] were also developed for resistance of UF membranes with a reported performance of average absolute error of 10%.

Data-driven models of membrane desalination (NF and RO) have been proposed to describe various aspects of steady state process performance with respect to salt rejection [16, 22, 24, 25], permeate flux [24, 26], as well as modeling of membrane fouling [8, 27, 28]. For example, back-propagation ANN models were used [25] to model the rejection of NaCl and $MgCl_2$ salts (based on laboratory scale steady state NF plant data for feed concentration of 5000-25,000 mg/L) demonstrating average absolute deviation of 5%. In

the above work, models of different ANN architectures and training algorithms were assessed for two different sets of input variables (feed pressure or permeate flux and feed salt concentration) with salt rejection as the output variable. More recently, an interesting approach to modeling steady state RO plant performance was proposed in which the use of the product of salt rejection and permeate flux was introduced as an index of plant performance [24]. Using data for spiral-wound RO desalting of aqueous sodium chloride solutions, an ANN model was developed (input parameters included salt concentration, feed temperature, feed flow rate and feed pressure) for the plant performance index which demonstrated higher performance for salt concentrations of 6000 mg/L and 30,000 mg/L.

The majority of efforts on the development of data-driven RO process models have focused on the use of ANN algorithms given their ability to describe complex non-linear behavior [34]. However, such models require optimal ANN architecture while avoiding over-fitting and convergence to local minima [29]. Support Vector Machine (SVM) algorithm is an alternative method for developing data-driven models since it is based on the Structural Risk Minimization principle and thus avoids the convergence to local minima, while avoiding over fitting through control of the number of support vectors [30]. The use of SVM is especially useful for developing non-linear controllers as has been demonstrated in recent studies involving membrane based and other industrial processes [31-33]. For example, SVM based non-linear predictive functional control design was applied to a coking furnace, improving the regulatory capacity for both reference input tracking and load disturbance rejection compared with traditional PFC and PID control strategies [31]. A Least squares (LS)-SVM model was shown to be effective for developing a non-linear temperature controller for a proton exchange membrane fuel cell plant [32, 34]. SVM, in addition to radial basis function (RBF)-based ANN, was also reported effective in developing a data-driven model [33] of fouling of a membrane bioreactor (quantified via flux decline) making use of eight input parameters (e.g., membrane aperture, aeration gas quantity, initial membrane flux, operating pressure, water temperature, pumping time, sludge concentration and sludge granule). SVM as well as back-propagation ANN algorithms were also applied to developing forecasting models of brackish water RO plant performance [35], with respect to permeate flow rate and salt passage, where variability of up to 25% and 10% was experienced with respect to the normalized permeate flux and salt

rejection, respectively. It was shown, that time-series ANN based models enabled forecasting of salt passage and permeate flow rate up to 24 h with similar prediction errors for the SVM and ANN models (average absolute relative errors of 1.2% and 6.6%, respectively). The above models while suitable for long-term plant response (order of hours and above), do not capture short-time scale dynamic responses (order of seconds to minutes) of the system (e.g., due to sudden changes in input pressure or feed flow rate) which would be necessary, for example, for real-time plant control and fault detection.

Data-driven models could be particularly useful for use in plant controllers, identification of deviation of plant behavior from the expected norm, for sensor fault detection and even for smoothing of fluctuations in sensor data. However, such models must be able to accurately describe plant performance not only under steady state conditions, but more importantly under unsteady state operation and over time scales that capture short-time transients. Accordingly, the present work presents an approach for the development and integration of both steady and unsteady state data-driven plant models of RO desalting based on support vector regression (SVR). It is shown that SVR models can accurately describe RO plant performance (e.g., permeate and retentate flow rates and their respective salinities) based on basic operational plant parameters (i.e., feed flow rate, feed pressure and feed conductivity). Moreover, data-driven models for transient plant operation can provide accurate performance forecasting that is suitable for fault-tolerant control of RO plants.

2. Experimental procedure

2.1 Feed solution and materials

Aqueous salt feed solutions were prepared using analytical grade sodium chloride (Fisher Scientific, ACS grade, Pittsburgh, Pennsylvania) in deionized (DI) water. Solutions of two different salt concentrations were utilized (7500 and 5000 mg/L) with the feed solutions maintained at pH~7. Spiral-wound RO membranes that were utilized in pilot RO system (Dow Filmtec XLE-2540, The Dow Chemical Company, Midland, Michigan) were 2.5 inch

(outer diameter) 40 inch long elements (0.0635 m and 1.02 m, respectively) with an average surface area of 2.6 m^2. A single membrane element had a manufacturer reported permeate flow rate of 3.2 m^3/day, and a salt rejection of 99%, as determined at a pressure of 6.9 bar for a 500 mg/L NaCl solution. Each membrane element was contained in a separate pressure vessel with six membranes connected in series.

2.2 Description of experimental equipment

Data for model development were generated using the UCLA spiral-wound Mini-Mobile-Modular (M3) pilot RO desalination system shown schematically in **Fig. 1** [3, 36, 37]. The M3 system was designed for permeate water production capacity up to 1.2 m^3/h (7560 gallons/day) for brackish water (5000 mg/L TDS) operating at 75% recovery and up to 0.64 m^3/h (4058 gallons/day) for seawater desalination (at recovery of 40%) using up to 18 spiral-wound elements in various configurations. In the present study, a configuration of six elements in series was utilized with the system operating in a total recycle mode with the permeate and concentrate streams returned to the feed tank. Briefly, the M3 RO plant consisted of a 450 L feed tank with two low-pressure feed pumps (Model JM3460-SRM, Sea Recovery, Carson, CA) pumping the RO feed through a series of cartridge microfilters (5 µm, 0.45 µm and 0.2 µm; 08P GIANT, pleated 177 polypropylene filter cartridges, Keystone Filter, Hatfield, PA). The M3 was operated such that the filtered feed was fed to the RO membranes via two-high pressure pumps (Danfoss Model CM 3559, 3HP, 3450RPM, Baldor Reliance Motor, Sea Recovery Corp. Carson, California) operating in parallel and controlled by variable frequency drives (VFDs) (Model FM50, TECO Fluxmaster, Round Rock, Texas). The retentate flow rate and pressure in the RO unit were set by a model-based controller [36] that adjusts both electrically actuated needle valve (valve V-1) (model VA8V-7-0-10, ETI Systems, Carlsbad, California) on the retentate stream of the M3 RO system and the pump VFD. In order to maintain the temperature of the RO feed, a heat exchanger (Model BP 410-030 Refrigerant heat exchanger, ITT Industries) was installed on the retentate side of the RO system. Permeate and retentate streams of the M3 were monitored in-line via conductivity sensors, conductivity/resistivity sensor electronics (Signet 2839 to 2842 and Signet 2850, George Fischer Signet, Inc. El

Monte, California) and pH sensor (DryLoc pH electrodes 2775, George Fischer Signet, Inc. El Monte, California). The conductivity meters were calibrated over the expected concentration range for the study. The M3 plant was equipped with a centralized data acquisition system which receives all sensor outputs (0-5 V, 0-10 V, 4-20 mA) which are then converted to process variable values. The data were logged into a local computer as well as onto a network database. Data could be logged at a frequency range as high as 1 kHz, although for the present study a frequency of a 1 Hz was employed.

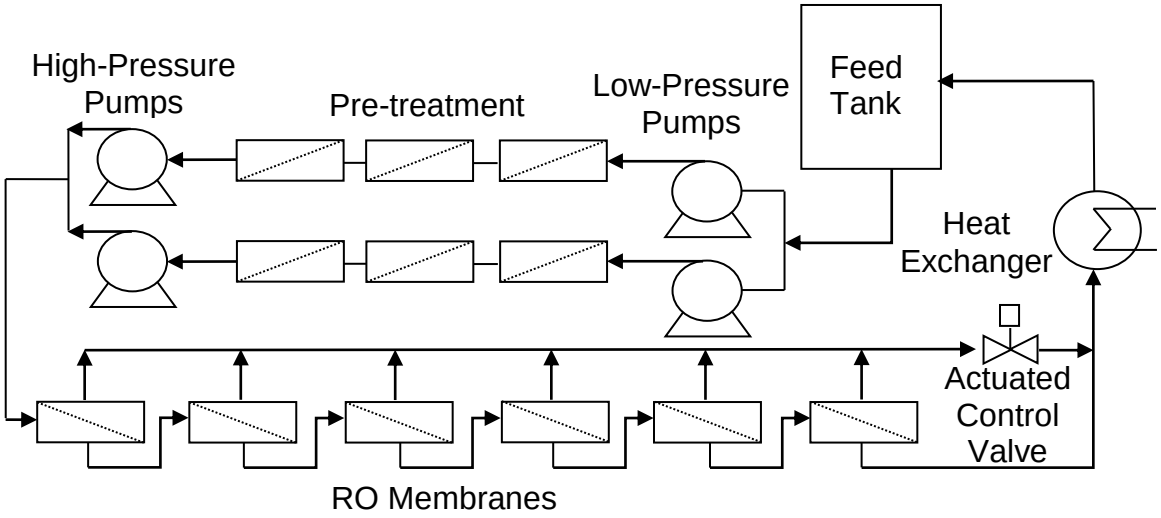

Figure 1. Configuration of the Spiral-wound RO pilot plant.

2.3 Experimental procedure

RO desalting experiments covered the range of operating conditions permitted by the operability limit of the M3 system for the specific feed salinities. The M3 control system was programmed to autonomously step through a range of feed flow rates and transmembrane pressures. Feed flow rate and pressure were varied by changing the speed of the high-pressure pumps and the actuated valve settings (**Fig. 1**). Feed pressure was varied by changing the actuated valve position while maintaining a constant feed flow rate (constant VFD), allowing the plant to operate until the attainment of steady state. The above experiments covered feed pressure and feed flow rate ranges of 6.9-26 bar (or 100-375 psi), and 0.23-0.68 m^3/h (or 1-3 gpm), respectively.

3. Model development

3.1 Data pre-processing

Operational M3 pilot data were acquired for both steady state operations, as well as for transient periods (except plant startup and shutdown), for which there were pressure and flow rate step changes of up to 15% and 25%, respectively. Operational parameters were recorded at a frequency of 1 Hz for each steady and during transitions between states. The recorded steady state data (**Table 1**) included natural plant fluctuations (i.e., due to operation of pumps and valves) as well as noise from normal operation of sensors, actuators and system pumps. However, in developing the data-driven models data smoothing was not employed given the small variance of the noise, and the objective of developing a simulator for the actual plant operation.

Table 1. Relative standard deviation for the range of steady state plant operating parameters covered in the study[a].

Variable	Minimum	Maximum	Av. STDV (%)
Feed Flow rate (m^3/h)	0.26	0.67	0.85
Feed Conductivity (µS)	9842	15828	0.44
Feed Pressure (bar)	8.13	24.62	0.43
Retentate Flow rate (m^3/h)	0.14	0.53	0.80
Retentate Conductivity (µS)	11,539	24,907	0.29
Retentate Pressure (bar)	3.59	11.29	0.99
Permeate Flow rate (m^3/h)	0.03	0.28	1.66
Permeate Conductivity (µS)	628.58	4024	0.44

[a] The Av.STDV= $\left(\sum_{i=1}^{N} STDV_i / V_{i,ave} \right) / N$ where STDV is the standard deviation for the given variable within a steady state trace i, $V_{i,ave}$ is the average variable value for the given trace, and N is the number of the experimental steady state traces (or experiments). Note: The salinity conversion factors for the permeate and retentate streams were $S=4.337 \cdot 10^{-4} \cdot C^{1.0201}$ and $S=3.833 \cdot 10^{-4} \cdot C^{1.0391}$ respectively, where S is salinity (mg/L total dissolved solids) and C is conductivity (µS).

Data-driven models of the state-the-plant (i.e., steady state) for permeate and retentate flow rates and conductivities were first developed from recorded data selected at 0.1 Hz sampling frequency in order to accelerate model training. The steady state period was established as that for which the measured process variables did not vary with time by more than 3% with respect to the time-averaged values. In all cases only the last five minute trace from the steady state period was utilized. Models for transient operation describing the evolution of permeate and retentate conductivities were developed based on data acquired from experiments in which the operating conditions were perturbed from steady state. Steady state was generally reached within a period of about 10 min after the perturbation. Higher frequency captured data (0.5 Hz) were utilized for modeling transient operation. Although data were logged by the M3 at an even higher frequency, use of higher frequency data would increase data redundancy and correspondingly the computational time for model training.

Data for model developments were normalized in the range of [0, 1] using min-max normalization:

$$y_i^{(n)} = \frac{y_i - \min(y)}{\max(y) - \min(y)} \quad (1)$$

where $y_i^{(n)}$ is the normalized value of the experimental data (y_i) and *min(y)* and *max(y)* are the minimum and maximum variable values in the data set. Subsequently, the data were divided into two sets, one for model training and the other for model testing. Data selected for training were used to adjust the model while test data (i.e., data that have not been used for model development) were used to evaluate model performance. In order to obtain state-of-the-plant models with good generalization capability, the training dataset was selected to cover the entire plant operational domain (i.e., applicability domain; [40]). Data for model testing were also selected within the applicability domain to avoid extrapolation during predictions. In all cases, complete steady state and transient sequences were selected for the training and test sets. For the transient models, self-organizing-maps (SOM) analysis [38] of the transient data was first performed whereby operational data of similar pattern were clustered in SOM cells. Training and test data for the transient models were then selected

from the different cells in order to ensure that adequate data representation is achieved for the whole operational domain.

3.2 Support vector regression (SVR) models

Since the relationships among process variables in the RO process are highly non-linear, both the steady state and unsteady state plant performance models (**Sections 3.3 and 3.4**) were developed using the support vector regression (SVR) algorithm [39, 40]. Briefly, given a vector x of RO process variables (e.g., feed pressure, feed flow rate, and feed conductivity), the goal of SVR is to find a function $f(x)$ that has at most ε deviation with respect to the actual values of a given target RO process variable y (e.g., permeate conductivity) and at the same time is as flat as possible (**Fig. 2**). SVR can be formulated as a convex optimization problem where a set of coefficients w for the regression model are computed in such a way that the flatness and the accuracy of $f(x)$ are maximized. Since it is not always possible to keep the error within the margin ε for all the available data points, a pair of slack variables ξ and ξ^* must be introduced within the SVR formulation to cope with otherwise infeasible constraints in the optimization (Fig. 2). In most cases, the optimization problem can be solved more readily by projecting x (i.e., the vector of the input RO process variables) onto a higher-dimensional space where linear regression models can be developed for the target RO process variable of interest. The functions used to perform the above linear to non-linear mapping are known as kernel functions. Finally, non-linear models relating input and target variables can be obtained by mapping the data back to its original (i.e., non-linear) space of RO process variables. In the SVR formulation only a subset of the training data points, representing the overall data behavior (i.e. support vectors), are used to generate the regression model (**Fig. 2**).

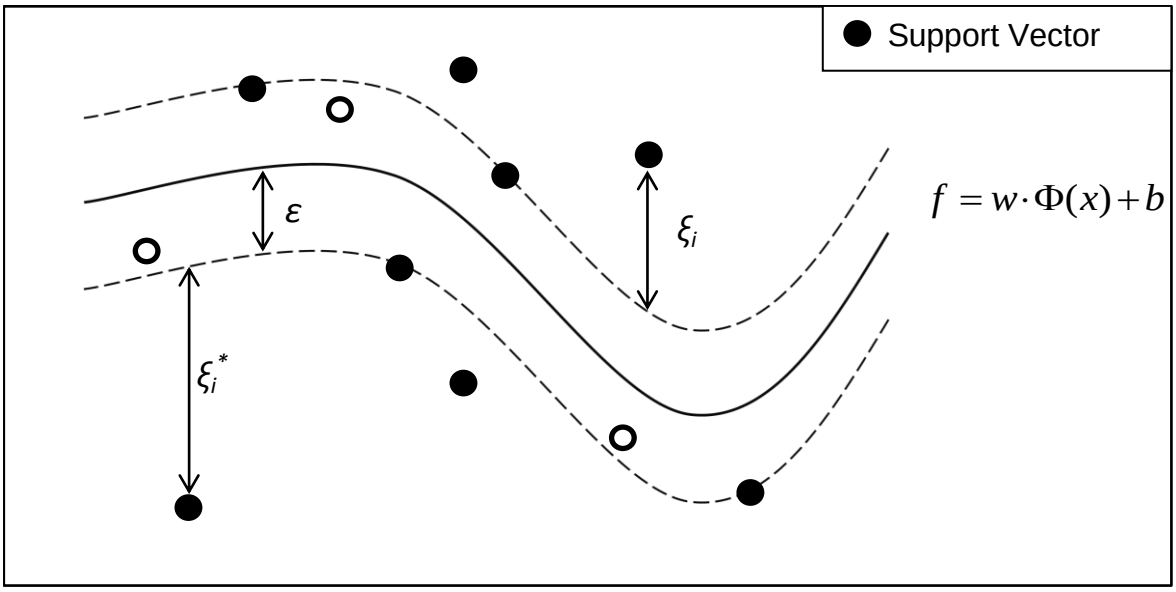

Figure 2. Support vector regression structural parameters. Regression function *f* supported on the most representative vectors of information (support vectors), *w* the normal vector to the hyperplane generated, the kernel function Φ, the vector of biases *b*, slack variables ξ and radius of the insensitive tube ε.

In the current work, the kernel function used for model development was the radial basis function (RBF) which is suitable for highly non-linear behaviors [40]. The parameter characterizing this kernel is the width of the Gaussian, σ, which determines the area of influence of the support vectors over the data space and here its optimal value was determined via grid search. The SVR based models were developed in MATLAB using the LS-SVMlab1.7 package [41, 42]. This SVR implementation utilized a regularization parameter γ, which controls the tradeoff between the flatness (or smoothness) of the models and their accuracy, whose optimal value was also determined via a grid search. **Table 2** summarizes the optimal values of σ and γ obtained via grid optimization for each model.

The performance of the different models (**Sections 3.3 and 3.4**) was quantified using the linear r^2 correlation coefficient (between the predicted, y_i^*, and experimental, y_i, variables) and by the average absolute (AAE) and average absolute relative (AARE) errors defined as:

$$AAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i^* - y_i\right| \qquad (2)$$

$$AARE = \frac{1}{n}\sum_{i=1}^{n}\frac{\left|y_i^* - y_i\right|}{y_i} \times 100 \quad (3)$$

where n is the total number of data points.

3.3 State-of-the-Plant Models

State-of-the-plant (STP) models were developed for steady state operation using feed flow rate, feed conductivity and feed pressure as input parameters. The data for the steady state segments (for the range of operating conditions listed in **Table 1**) were divided into training (60% of data, 2394 samples) and test (40% of data, 1596 samples) sets. Two different modeling approaches were implemented. First, individual models (**Eq. 4**) were constructed for each of the four target output variables (permeate flow rate, permeate conductivity, retentate flow rate and retentate conductivity), followed by a composite output parameters (OP) model to simultaneously predict the four output variables with the same three input parameters (**Eq. 5**),

$$F_i = f(C_f, Q_f, P_f) \quad (4)$$

$$\left[C_p, Q_p, C_r, Q_r\right] = f(C_f, Q_f, P_f) \quad (5)$$

where F_i is one of (permeate flow rate, permeate conductivity, retentate flow rate, or retentate conductivity), and C, Q and P are conductivity, flow rate and pressure, respectively, and the subscripts f, p and r refer to feed, permeate and retentate streams, respectively. For the individual OP models the optimal value of γ was 4200 for all the target variables. The optimal value of σ was 12 for the retentate flow rate model and 8.8 for the permeate flow rate, permeate conductivity and retentate conductivity models. The optimal γ and σ parameters for the composite OP model were 1000 and 7, respectively.

3.4 Unsteady State Models

Data-driven unsteady state plant models for the conductivity of the retentate and permeate streams were developed using a modeling structure that considers a time marching approach (**Fig. 3**).

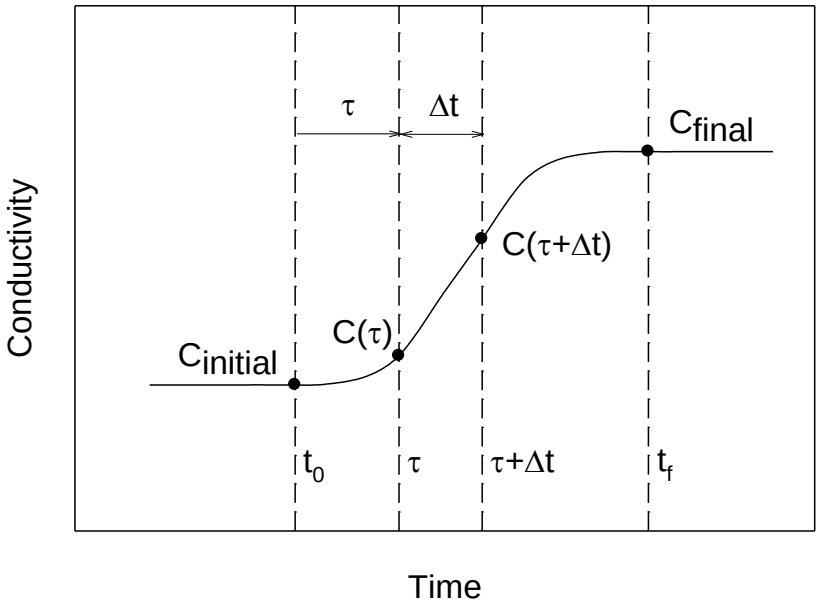

Figure 3. Illustration of a transient conductivity trace with input and output parameters used in Eq. (6a) for predicting the evolution of conductivity between the initial and final steady states ($C_{initial}$ and C_{final}) at times t_0 and t_f, respectively. The parameters τ and $\tau + \Delta t$ refer to the elapsed time relative to t_o and forecasting period, respectively.

In this approach, the data-driven transient model is constructed to enable a marching forecasting prediction of stream conductivities at a prescribed period of time, Δt, forward of elapsed time τ from the change in operating conditions at t_0 (i.e., $\tau = t - t_o$). Accordingly, the transient model is expressed as

$$\left[C(\tau + \Delta t)\right] = f\left(C(\tau), C_{initial}, C_{final}, \tau, \Delta t\right) \quad (6a)$$

in which steady state data for the modeled stream conductivities, prior to and post the perturbation of the steady state operating condition, are input variables that are in turn predicted from the steady state plant model (**Section 3.3, Eq. 5**)

$$\left[C_{initial}, C_{final}\right] = f\left(C_f, Q_f, P_f\right) \quad (6b)$$

in which $C_{initial}$ and C_{final} are the stream conductivities at time t_o and t_f, at the given initial steady state operation and the new steady state, respectively. The optimal γ and σ parameters for building the SVR composite OP model were 10,000 and 18, respectively. Finally, the transient models were developed based on a dataset of 122 different unsteady state runs (60 were utilized for model testing), covering the range of operating conditions given in **Table 1**, in which either pressure and flow rates were perturbed by up to 15% and 25%, respectively.

4. Results and discussion

4.1 State-of-the-plant model

State-of-the-plant (STP) model (**Section 3.3**) was first developed in order to establish the base model for steady state operation of the RO plant (**Section 2.3**). Performances of the individual and composite OP models are provided in **Table 2**. Individual OP models predicted permeate and retentate flow rates with average absolute errors (AAEs) of 0.013 m^3/h (2.46%) and 0.011 m^3/h (0.70%), respectively, for permeate and retentate flow rate ranges of 0.023-0.295 m^3/h (0.1-1.3 gpm) and 0.136 - 0.522 m^3/h (0.6-2.3 gpm), respectively. Similarly, permeate and retentate conductivities were predicted with AAE levels of 21.7 μS (1.24% AARE) and 110.2 μS (0.75% AARE) (**Table 2**), respectively, for corresponding conductivity ranges of 630-4000 μS and 11,500-25,000 μS. The correlation coefficient (r^2) for the linear correlation (predicted versus measurements) was in all cases ≥ 0.994.

Table 2. Performance of SVR based models for predicting steady state RO plant performance based on individual single output parameter models and a composite model. [a]

Model	Predicted Variable	γ	σ	AAE	AARE (%)	r^2
Individual	Permeate Flow rate (m^3/h)	4200	8.8	0.013	2.46	0.994
Individual	Permeate Conductivity (μS)	4200	8.8	21.72	1.24	0.999
Individual	Retentate Flow rate (m^3/h)	4200	12	0.011	0.70	0.998
Individual	Retentate Conductivity (μS)	4200	8.8	110.16	0.75	0.998
Composite	Permeate Flow rate (m^3/h)	1000	7	0.011	2.54	0.993
	Permeate Conductivity (μS)			22.74	1.32	0.999
	Retentate Flow rate (m^3/h)			0.011	0.71	0.998
	Retentate Conductivity (μS)			117.01	0.80	0.997

[a] γ - regularization parameter; σ - width of the Gaussian; AAE and AARE are the average absolute and average absolute relative errors, respectively, and r^2 is the linear correlation coefficient.

The composite OP model provided simultaneous prediction of all four output parameters at a similar level of accuracy (**Table 2; Figs. 4**). Permeate and retentate flow rates (**Fig. 4a and b,** respectively) were predicted with average absolute errors of 0.011 m^3/h (corresponding to 2.54% and 0.71% AARE for the above two streams). Conductivities of the permeate and retentate stream (**Fig. 4c and d** respectively) were predicted with AAE values of 22.7 μS (1.32% AARE) and 117.0 μS (0.80% AARE), respectively (**Table 2**). The r^2 linear correlation coefficient was ≥ 0.993 for the composite OP model. For both the individual and composite OP models, AARE values for the predictions were greater for the permeate flow rate and conductivity relative to predictions of these parameters for the retentate stream. This trend is not surprising given the higher absolute values of the above parameters for the retentate stream.

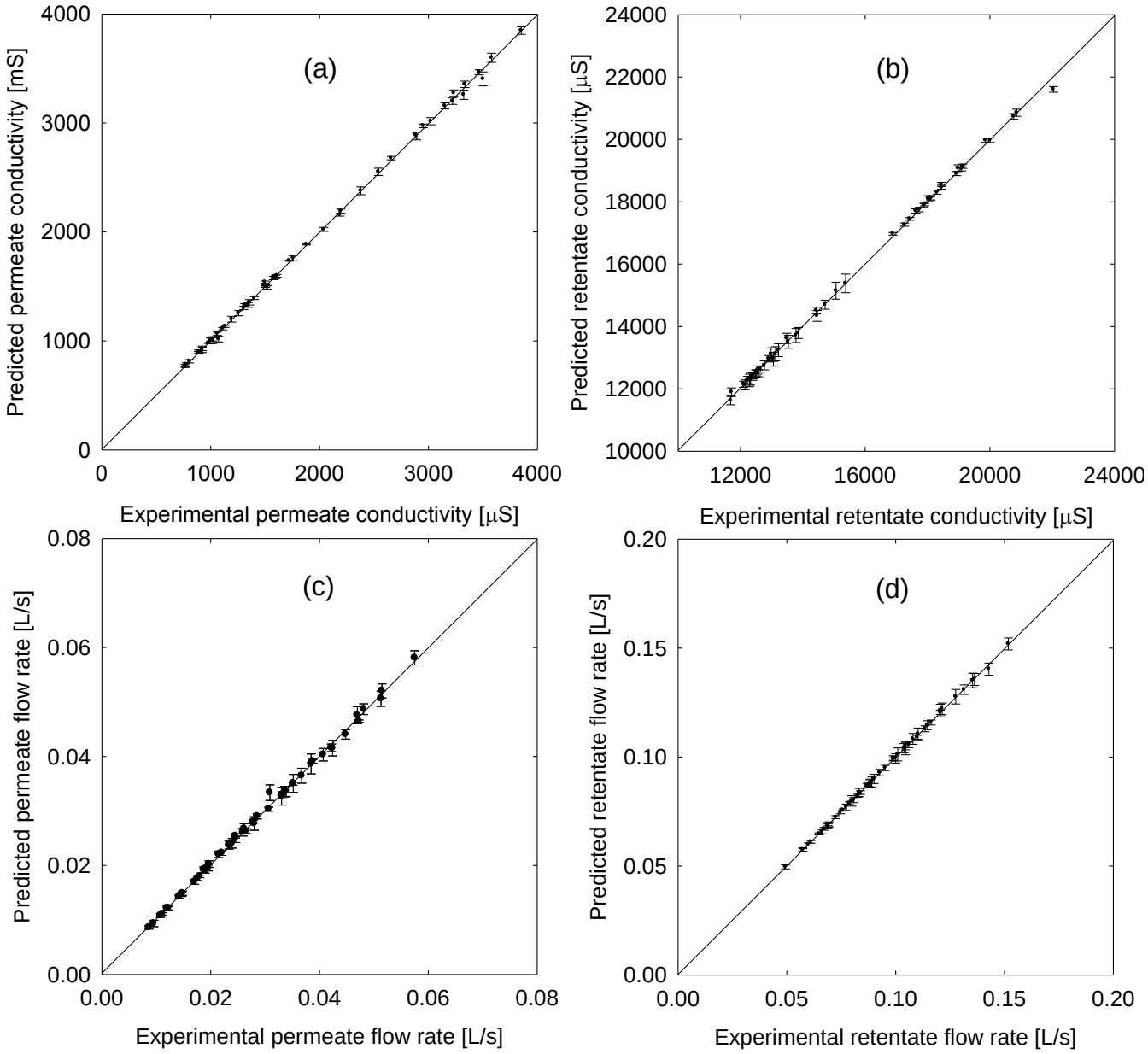

Figure 4. Comparison of experimental and composite model predictions for: (a) permeate, and (b) retentate conductivities; and (c) permeate and (d) retentate flow rates. Note: In order to maintain clarity of presentation only the predicted averages are plotted (along with a bar depicting the standard deviation) for each steady state data trace.

The difference in performance of the composite and individual OP models was on average less than 0.05% (**Table 2**). Overall, the AARE for the steady state individual and composite OP models were for the permeate flow rate (2.46% and 2.54%, respectively). The above performance level suggests that there is little advantage to using the individual

OP models relative to the composite OP model, particularly given that the latter is more convenient. Compared with the present SVR models, predictions obtained using the commercial ROSA software [12] (suitable for the present membrane elements, **Section 2.1**) are of lower accuracy, e.g., with AARE of 23% and 27% for the salinity of the permeate and retentate streams, respectively. The retentate and permeate flow rates were underpredicted and overpredicted by ROSA (AARE of 19% and 68%, respectively). The above behavior is not surprising since ROSA predictions were based on the manufacturer membrane permeability and without consideration of the efficiency of various plant components. In contrast, the SVR models were based on the actual experimental performance data and thus such data-driven models can serve as more reliable plant-specific operational models.

4.2 Transient operation models

Data-driven models of unsteady state (or transient) plant operation can be useful in refining plant operational control strategies by enabling prediction of the dynamic response of the plant post perturbation from steady state operation. In this regard, it is noted that plant hydraulic response is generally fast. In the present RO plant, flow rate transients were typically of the order of a few seconds relative to much longer system response (order of minutes) with respect to transients of retentate and permeate conductivities. Accordingly, from the perspective of plant control, state-of-the-plant models may be sufficiently effective for predicting the retentate and permeate flow rates upon pressure or feed flow rate transitions. In contrast, the dynamics of the conductivities of these streams (e.g., due to feed flow rate, feed salinity and pressure perturbations) are influenced by salt dispersion through the system and stabilization of salt flux through the membrane. For example, for the present RO pilot, transient periods of up to 10 min were required to reach steady state, with respect to salinity of the permeate and retentate streams, as a result of feed pressure perturbations.

Several data-driven models were developed to predict the transient response of permeate and retentate conductivities at different forecasting periods (i.e., Δt , **Section 3.4, Eq. 6a**) in the range of 2 to 210 s (**Table 3**). Longer forecasting periods (i.e., > 210 s) were not

considered since model accuracy decreased significantly due to the reduction of available training data with increasing Δt (see **Fig. 3**). Illustrations of model tracking of the measured permeate conductivities, for a set of different transient trajectories (induced by 5-15% perturbation of the applied pressure), are provided in **Fig. 5**, for model forecasting period (i.e., Δt) of 30 s. Forecasting at $\Delta t = 30\,\text{s}$ (**Fig. 5**) was highly accurate, for a range of different dynamics, with average AAE and AARE for the above transients being 6.16 μS and 0.47%, respectively. The average AAE corresponding to the retentate conductivity (**Table 3**) at the same conditions was slightly higher with a value of 38.12 μS (0.27% AARE). At the longer forecasting period of 120 s, permeate conductivity AAE and AARE were 9.54 μS and 0.66% respectively. Errors for retentate conductivity predictions were somewhat higher with AAE in the range of 50.85 μS (0.36% AARE).

Table 3. Performance of the transient operation models for prediction of permeate and retentate conductivities for different forecasting periods.

	Permeate			Retentate		
Δt (s)[(a)]	AAE (μS)	AARE (%)	r^2	AAE (μS)	AARE (%)	r^2
2	1.55	0.12	1	8.98	0.07	1
30	6.16	0.47	0.999	38.12	0.27	0.999
60	7.91	0.58	0.999	44.86	0.32	0.999
80	8.49	0.60	0.999	47.30	0.34	0.999
90	8.72	0.62	0.999	48.62	0.34	0.999
120	9.54	0.66	0.999	50.85	0.36	0.999
180	10.91	0.73	0.999	51.87	0.37	0.999
210	11.58	0.76	0.999	54.21	0.38	0.999

(a) Forecasting period Δt, ($t=\tau+\Delta t$; **Eq.6a**)

As expected, forecasting errors for conductivities increased (**Table 3**), for both the permeate and retentate streams, as the amount data available for model training decreased with increasing forecasting period (**Fig. 3**). The above trend is depicted in **Fig. 6** for both the permeate and retentate conductivities. For the forecasting period of Δt =2-210 s, errors for predictions of the permeate and retentate streams were in the range of AAE=1.55-11.58 μS (AARE 0.12-0.76%) and AAE=8.98-54.21 μS (AARE 0.07-0.38%), respectively (**Table 3**). Forecasting at very short times (e.g., $\Delta t = 2\,\text{s}$) is likely to be of limited value for plant control since differences in the transient data are very small over such short time intervals. However, predictions at longer forecasting periods (i.e., larger Δt) of up to 210 s (**Fig. 6**, **Table 3**) were excellent with an AARE of 0.44% and 0.23% for the conductivity of

the permeate and retentate streams, respectively. The somewhat lower AARE for the retentate conductivity predictions can be attributed to the greater accuracy of the state-of-the-plant models for predicting $C_{initial}$ and C_{final} (**Fig. 3**) for the retentate (0.8% AARE) relative to the permeate (1.32% AARE) streams. However transient predictions for both streams were clearly of reasonable comparable accuracy.

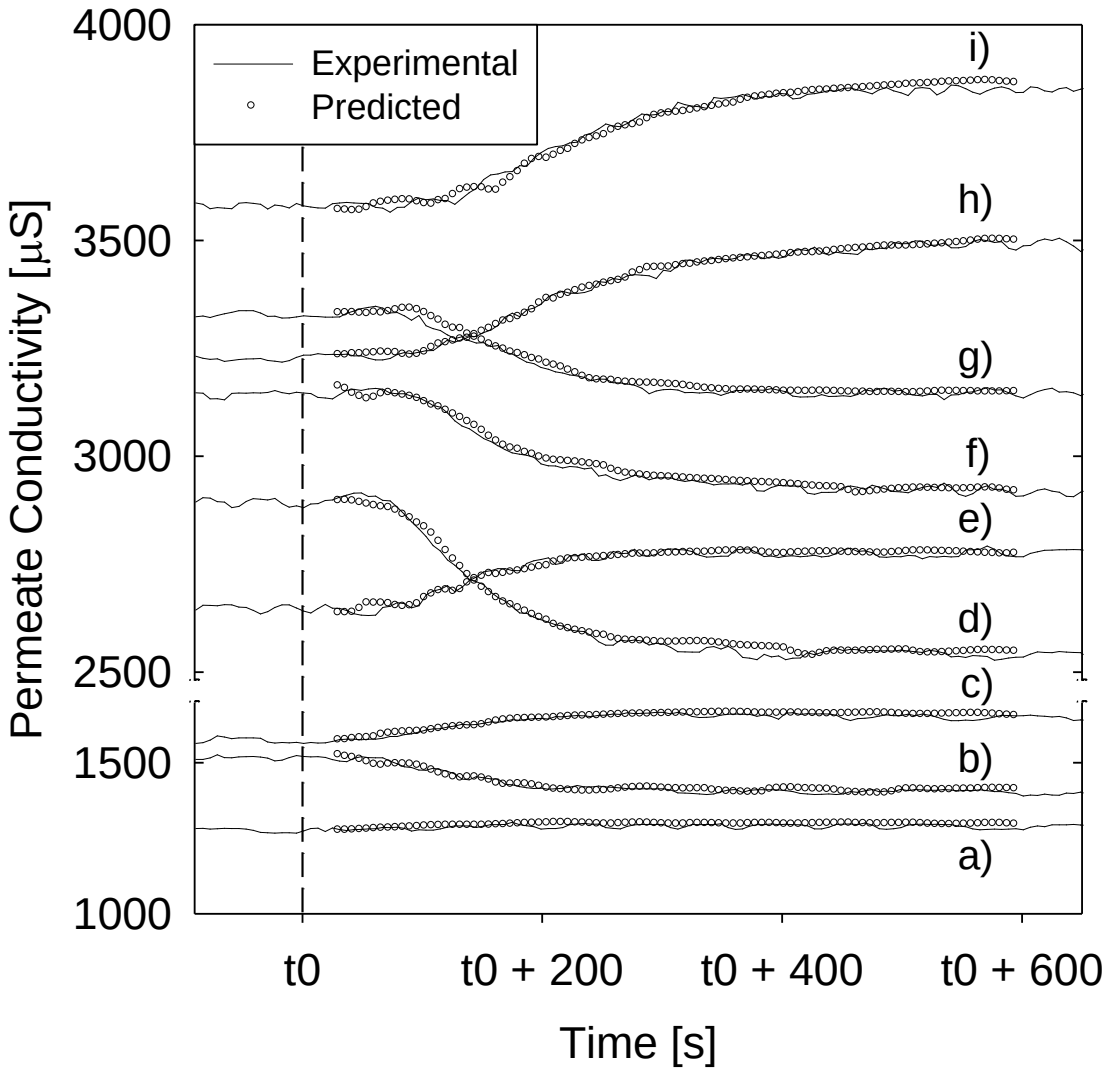

Figure 5. Comparison of transient model predictions and experimental permeate conductivity measurements for forecasting period of 30 s (Note: t0 denotes the starting time for the change in operating conditions). Unsteady state operation was induced by the following changes in either applied pressure or feed flow rate: a) 10.2-9.6 bar, 0.27 m^3/h, b) 9.6-9.2 bar, 0.31 m^3/h, c) 10.1-10.5 bar, 0.31 m^3/h, d) 11.8-12.2, 0.40 m^3/h, e) 12.0-11.6 bar, 0.35 m^3/h, f) 13.7-14.4 bar, 0.49 m^3/h, g) 11.0-10.5 bar, 0.36 m^3/h, h) 9.8-10.9 bar, 0.35-0.41 m^3/h, and i) 8.8-8.5 bar, 0.28 m^3/h.

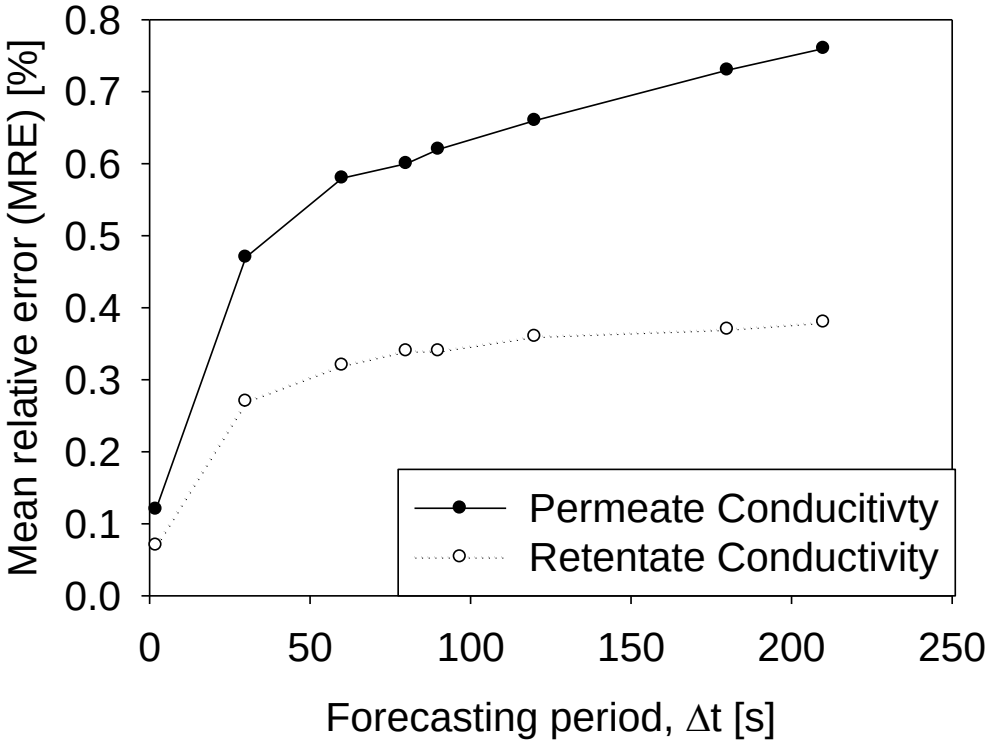

Figure 6. Performance of the unsteady state model for forecasting of permeate and retentate conductivities as a function of the forecasting period (Δt) averaged for each forecasting period over the transient test traces for the range of operating conditions given in **Table 1.**

5. Conclusions

The development of data-driven RO plant performance models was demonstrated using the support vector regression model building approach. Models of both steady state and unsteady state plant operation were developed based on a wide range of operational data obtained from a fully automated small spiral-wound RO pilot. Single output variable steady state plant models for flow rates and conductivities of the permeate and retentate streams were of high accuracy. Performance of a composite SVR based model (for both streams) for flow rates and conductivities was of comparable accuracy to the single output variable models. Predictions of stream conductivities for transient operation was achieved, making use of both predictions from state-of-the-plant model along with a performance time forecasting approach, with less than 1% AARE for forecasting periods in the range of 2-210 s. This level of accuracy suggests that short-term performance forecasting, based on plant performance data, can be particularly useful for the development of advanced RO plant control and for process optimization. For example, plant controllers can utilize data-driven models to aid for smoothing operational fluctuations, supplementing missing sensor data (e.g., due to sensor faults, short-term interruption in communication between sensors and data acquisition systems and/or controllers), and for fault tolerant control. Admittedly, data-driven models are plant specific and must be developed and applied for the desired applicability domain. In this regard, online model training can be implemented to improve model performance and its applicability domain as new plant data are acquired.

Acknowledgements

This work was supported in part by funding from the Catalan Government (2009 SGR 1529), the Spanish Ministry of Education (CTQ2009-14627), the California Department of Water Resources, the United States Environmental Protection Agency, the National Water Research Institute, the Office of Naval Research, and the UCLA Water Technology Research Center.

Dipòsit Legal: T 961-2014

Nomenclature

AAE	average absolute error
AARE	average absolute relative error
b	bias vector
C_f	feed conductivity
C_{final}	conductivity at the final steady state
$C_{initial}$	conductivity at the initial steady state
C_p	permeate conductivity
C_r	retentate conductivity
$C(\tau)$	conductivity at $t_0+\tau$
$C(\tau+\Delta t)$	predicted stream conductivity
f(x)	function obtained by SVR
n	number of experimental values
P_f	feed pressure
Q_f	feed flow rate
Q_p	permeate flow rate
Q_r	retentate flow rate
r^2	correlation coefficient
STDV	standard deviation
t_0	time at the change in the operating conditions
t_f	time at the new steady state

V_i	average variable value for the given steady state
w	normal vector to hyperplane
x	vector of input RO parameters
y_i	experimental value for a given variable
y_i^*	predicted value for a given variable
$y_i^{(n)}$	normalized value for a given variable
Δt	forecasting period
ε	tolerance of the SVR models
γ	regularization parameter
ξ, ξ^*	upper and lower slack variables
σ	width of the Gaussian
τ	elapsed time relative to t_0
Φ	kernel function

References

[1] K. Jamal, M.A. Khan, M. Kamil, Mathematical modeling of reverse osmosis systems, Desalination, 160 (2004) 29-42.

[2] A. Abbas, Model predictive control of a reverse osmosis desalination unit, Desalination, 194 (2006) 268-280.

[3] A.R. Bartman, P.D. Christofides, Y. Cohen, Nonlinear Model-Based Control of an Experimental Reverse-Osmosis Water Desalination System, Industrial & Engineering Chemistry Research, 48 (2009) 6126-6136.

[4] A.R. Bartman, C.W. McFall, P.D. Christofides, Y. Cohen, Model-predictive control of feed flow reversal in a reverse osmosis desalination process, Journal of Process Control, 19 (2009) 433-442.

[5] S.S. Sablani, M.F.A. Goosen, R. Al-Belushi, M. Wilf, Concentration polarization in ultrafiltration and reverse osmosis: a critical review, Desalination, 141 (2001) 269-289.

[6] G. Belfort, R.H. Davis, A.L. Zydney, The behavior of suspensions and macromolecular solutions in crossflow microfiltration, Journal of Membrane Science, 96 (1994) 1-58.

[7] S. Shirazi, C.-J. Lin, D. Chen, Inorganic fouling of pressure-driven membrane processes — A critical review, Desalination, 250 (2010) 236-248.

[8] Q.-F. Liu, S.-H. Kim, Evaluation of membrane fouling models based on bench-scale experiments: A comparison between constant flowrate blocking laws and artificial neural network (ANNs) model, Journal of Membrane Science, 310 (2008) 393-401.

[9] S. Gray, R. Semiat, M. Duke, A. Rahardianto, Y. Cohen, 4.04 - Seawater Use and Desalination Technology, in: W. Editor-in-Chief: Peter (Ed.) Treatise on Water Science, Elsevier, Oxford, 2011, pp. 73-109.

[10] Windflows, GE power & water, 2010.

[11] CAROL, Toray Membrane Europe AG,. Münchenstein, Switzerland, 2010.

[12] ROSA, Reverse Osmosis System Analysis, The Dow Chemical Company, 2010.

[13] J.S. Vrouwenvelder, C. Picioreanu, J.C. Kruithof, M.C.M. van Loosdrecht, Biofouling in spiral wound membrane systems: Three-dimensional CFD model based evaluation of experimental data, Journal of Membrane Science, 346 (2010) 71-85.

[14] K.L. Chen, L. Song, S.L. Ong, W.J. Ng, The development of membrane fouling in full-scale RO processes, Journal of Membrane Science, 232 (2004) 63-72.

[15] E. Alhseinat, R. Sheikholeslami, A completely theoretical approach for assessing fouling propensity along a full-scale reverse osmosis process, Desalination, 301 (2012) 1-9.

[16] W. Richard Bowen, M.G. Jones, J.S. Welfoot, H.N.S. Yousef, Predicting salt rejections at nanofiltration membranes using artificial neural networks, Desalination, 129 (2000) 147-162.

[17] J.L. Dirion, M. Cabassud, M.V. Le Lann, G. Casamatta, Development of adaptive neural networks for flexible control of batch processes, The Chemical Engineering Journal and the Biochemical Engineering Journal, 63 (1996) 65-77.

[18] W.T. Miller, R.S. Sutton, P.J. Werbos, Neural networks for control, Cambridge, MA, 1995.

[19] R. Rallo, J. Ferre-Giné, A. Arenas, F. Giralt, Neural virtual sensor for the inferential prediction of product quality from process variables, Computers & Chemical Engineering, 26 (2002) 1735-1754.

[20] M. Azlan Hussain, Review of the applications of neural networks in chemical process control — simulation and online implementation, Artificial Intelligence in Engineering, 13 (1999) 55-68.

[21] A. Shahsavand, M.P. Chenar, Neural networks modeling of hollow fiber membrane processes, Journal of Membrane Science, 297 (2007) 59-73.

[22] H. Al-Zoubi, N. Hilal, N.A. Darwish, A.W. Mohammad, Rejection and modelling of sulphate and potassium salts by nanofiltration membranes: neural network and Spiegler–Kedem model, Desalination, 206 (2007) 42-60.

[23] N. Delgrange-Vincent, C. Cabassud, M. Cabassud, L. Durand-Bourlier, J.M. Laîné, Neural networks for long term prediction of fouling and backwash efficiency in ultrafiltration for drinking water production, Desalination, 131 (2000) 353-362.

[24] M. Khayet, C. Cojocaru, M. Essalhi, Artificial neural network modeling and response surface methodology of desalination by reverse osmosis, Journal of Membrane Science, 368 (2011) 202-214.

[25] N.A. Darwish, N. Hilal, H. Al-Zoubi, A.W. Mohammad, Neural Networks Simulation of the Filtration of Sodium Chloride and Magnesium Chloride Solutions Using Nanofiltration Membranes, Chemical Engineering Research and Design, 85 (2007) 417-430.

[26] A. Abbas, N. Al-Bastaki, Modeling of an RO water desalination unit using neural networks, Chemical Engineering Journal, 114 (2005) 139-143.

[27] V. Vapnik, S.E. Golowich, A.J. Smola, Support Vector Method for Function Approximation, Regression Estimation and Signal Processing, in: M. Mozer, M.I. Jordan, T. Petsche (Eds.) Advances in Neural Information Processing Systems 9 --- Proceedings of the 1996 Neural Information Processing Systems Conference (NIPS 1996), MIT Press, Cambridge, MA, USA, Dever, CO, USA, 1997, pp. 281-287.

[28] H. Drucker, C.J.C. Burges, L. Kaufman, A.J. Smola, V. Vapnik, Support vector regression machines, in: Advances in Neural Information Processing Systems 9, MIT Press, Cambridge, MA, USA, 1997, pp. 155-161.

[29] P. Bhagat, An introduction to neural nets, Chemical engineering progress, 86 (1990) 55-60.

[30] N. Cristianini, J. Shawe-Taylor, An introduction to support vector machines and other kernel-based learning methods, Cambridge University Press, 2000.

[31] R. Zhang, S. Wang, Support vector machine based predictive functional control design for output temperature of coking furnace, Journal of Process Control, 18 (2008) 439-448.

[32] X. Li, G.-y. Cao, X.-j. Zhu, Modeling and control of PEMFC based on least squares support vector machines, Energy Conversion and Management, 47 (2006) 1032-1050.

[33] G. Meijuan, T. Jingwen, L. Jin, The Study of Membrane Fouling Modeling Method Based on Support Vector Machine for Sewage Treatment Membrane Bioreactor, in: Industrial Electronics and Applications, 2007. ICIEA 2007. 2nd IEEE Conference on, 2007, pp. 1393-1398.

[34] R. Yuan, C. Guang-Yi, Z. Xin-Jian, Predictive control of proton exchange membrane fuel cell (PEMFC) based on support vector regression machine, in: Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on, 2005, pp. 4028-4031 Vol. 4027.

[35] D. Libotean, J. Giralt, F. Giralt, R. Rallo, T. Wolfe, Y. Cohen, Neural network approach for modeling the performance of reverse osmosis membrane desalting, Journal of Membrane Science, 326 (2009) 408-419.

[36] A.R. Bartman, A. Zhu, P.D. Christofides, Y. Cohen, Minimizing energy consumption in reverse osmosis membrane desalination using optimization-based control, Journal of Process Control, 20 (2010) 1261-1269.

[37] H. Gu, A.R. Bartman, M. Uchymiak, P.D. Christofides, Y. Cohen, Self-adaptive feed flow reversal operation of reverse osmosis desalination, Desalination.

[38] K.F. Thang, R.K. Aggarwal, A.J. McGrail, D.G. Esp, Analysis of power transformer dissolved gas data using the self-organizing map, Power Delivery, IEEE Transactions on, 18 (2003) 1241-1248.

[39] M. Vogt, V. Kecman, Active-set methods for support vector machines, in: L. Wang (Ed.) Support vector machines: Theory and applications, Springer, Berlin, 2005, pp. 133-158.

[40] C.M. Bishop, Pattern Recognition and Machine Learning, Springer Science+ Business Media, LLC, New York, NY, 2006.

[41] V.N. Vapnik, The nature of statistical learning theory, Springer-Verlag New York, Inc., 1995.

[42] K. De Brabanter, P. Karsmakers, F. Ojeda, C. Alzate, J. De Brabanter, K. Pelckmans, B. De Moor, J. Vandewalle, A.K. Suykens, LS-SVMlab Toolbox user's guide, in, K.U. Leuven Leuven, Belgium, 2011.

8. Article 2. Fault detection and isolation in a spiral-wound reverse osmosis (RO) desalination plant

Article 2

Authors: X. Pascual, H. Gu, A. Bartman, A. Zhu, A. Rahardianto, J. Giralt, R. Rallo, P. D. Christofides, Y. Cohen.

Title: Fault Detection and Isolation in a Spiral-Wound Reverse Osmosis (RO) Desalination Plant.

Journal: Industrial and Engineering Chemistry Research, ISSN: 0888-5885

In press

ISI category: Chemical Engineering **Impact factor:** 2.206

Position in the category: 30/133

Cites: -

Observations: Accepted DOI 10.1021/ie403603x

Abstract

Sensor fault detection and isolation (SFDI) approaches, based on Support vector regression (SVR) plant sensor models and Self-organizing-map (SOM) analysis, were investigated for application to reverse osmosis (RO) desalination plant operation. SFDI-SVR and SFDI-SOM were assessed using operational data from a small spiral-wound RO pilot plant and synthetic faulty data generated as perturbations relative to normal plant operational data. SFDI-SVR was achieved without false negative (FN) detections for sensor deviations $\gtrsim |10\%|$, and FN detections of at most $\lesssim |5\%|$ for sensor deviations $\gtrsim |4\%|$ at sensor fault detection (FD) thresholds of up to ~|4%|. False positive (FP) detections were nearly invariant with respect to sensor FD, being $\lesssim |5\%|$ for sensor deviations of $\gtrsim |5\%|$. Corrections of faulty sensor readings were within SVR model accuracy (AARE < 1%) for SFDI-SVR and $\lesssim |5\%|$ for SFDI-SOM. Although SFDI-SOM has lower detection accuracy, it requires a single overall plant model (or SOM), while providing pictorial representation of plant operation and depiction of faulty operational trajectories.

1. Introduction

Water desalination by reverse osmosis (RO) membrane technology has been increasingly deployed for potable water production from seawater and water reuse applications including municipal wastewater and agricultural drainage water. Most RO plants are generally designed to operate at relatively steady-state conditions with traditional control strategies to attain the target permeate productivity and quality. Given the complexity of RO plants, plant process models, which consider specific plant characteristics and equipment, are needed to describe plant operation in order to optimize water production and design robust process control strategies [1-4].

Plant process models require reliable sensor measurements as deviation of sensor readings (e.g., due to sensor failure or drift) can result in drift of plant operational variables beyond acceptable limits [5]. In this regard, fault tolerant control (FTC) [6] strategies need to be provided with effective fault detection and isolation (FDI) methods to identify faults in critical plant sensors or actuators that could degrade control system performance. FDI methods can be designed to detect system faults, and to identify their root cause by isolating system components whose operation lies outside the nominal range [7]. Indeed, model-based fault detection and isolation systems have been successfully applied in different fields, and have been integrated with systems of vehicle control, power, manufacturing, as well as in robotics and process control systems [6,8-13].

In recent years, model-based FDI methods have been applied in RO water desalination [9,14-16]. For example, actuator FDI integrated with a fault-tolerant-control (FTC) strategy was reported for a single membrane unit RO desalination process without pre-treatment or post-treatment. The approach [15] relied on model-based feedback control laws making use of the fundamental RO transport equations and was successfully applied to operation with varying levels of feed salinity fluctuations. A supervisory switching law was derived to guarantee closed-loop stability by determining the activation time of fall-back control configurations in the presence of faults in the primary control configuration. The above approach was able to detect and isolate actuator faults in the system's adjustable retentate and pump bypass valves, as well as to recover the desired system operational regime by switching to the appropriate control strategy (i.e., using redundant actuators different than

the actuators used in the primary control configuration). The above approach was feasible due, in part, to the deployment of redundant controls that provided alternatives to compensate for faulty elements [17-18].

Sensor fault detection and isolation (SFDI), also known as gross error detection or sensor validation[5] is a specific case of the general fault diagnosis and handling problem. Once a faulty sensor is detected and isolated, data reconciliation or rectification is necessary in order to estimate the sensor true reading values for the faulty data trace. The above approach was demonstrated for RO desalination plants for detection of sensor along with other faults in the plant [14, 16] as the basis for model-based fault tolerant RO plant control [14]. Faults (e.g., faults in system hydraulics, pumps, sensors and actuators) were induced in a computerized model of the RO plant (treatment capacity of 500 L/h of feed with a conductivity value of 800 µS/cm producing 250 L/h permeate of 7 µS/cm conductivity). Fault tolerant control was demonstrated [14] for two different scenarios, the first involving reduction of the retentate valve speed and a second in which a constant negative offset was applied to the permeate flow rate sensor. It should be noted that in the latter case the permeate flow rate sensor was considered faulty when balance closure errors exceeded a specific tolerance. Such an approach can be beneficial when there is *a priori* knowledge (i.e., isolation) of the faulty sensor. However, in actual plant operation, detection and isolation are crucial since one needs to identify (via appropriate procedures) and confirm which sensors are providing normal or abnormal (including missing) readings.

Statistical techniques such as Principal Component Analysis (PCA), based on the analysis of process history, are commonly used for fault detection and monitoring. For example, monitoring and fault detection was carried out applying a PCA-based scheme and an unfolded PCA (U-PCA) using an RO desalination plant simulation (~1.7 m^3/h capacity with sand and cartridge filters for RO feed pretreatment) that operated with cyclical cleaning phases [16]. The approach was tested by simulating plant operation using an object-oriented and dynamic simulation tool (EcosimPro [19]). Three different U-PCA models were developed (for the two filters and for the RO membranes). A PCA model was first established using nominal data, where the square prediction error (Q) was used for all plant variables (including pressures, flow rates and concentrations) to monitor and detect

faulty data. Three types of faults (generated as 0%-60% deviation from the nominal sensor readings) were considered in the plant that included offset in the sand filter pressure sensor, as well as various types of membrane blockages and integrity losses. Faults detection was established on the basis of *Q* values exceeding a specific threshold, and whereby the Trimmed Score Regression method (TSR) [20] was used for correction of faulty data. The number of false positives observed with the U-PCA method relative to classical PCA was reduced from 11.6%, 12.9% and 10.5% to 10.2%, 11.3% and 4.0% for the sand filter, cartridge filter and membrane models, respectively. Faults were detected (i.e., 100% detection) without delay (i.e., instantaneous detection), since only abrupt faults were considered. It is important to note that abrupt faults are only a subset of the faults that can occur in the operation of RO plants. For example, instrument behavior may deviate gradually from its nominal operation (for example during fouling, scaling and sensor drifting). In such cases additional strategies that take into account historical plant operational data are essential to confirm fault identification. However, detection of the above would clearly involve a time delay due to the nature of such faults.

The majority of previous studies have focused on FDI of equipment and actuators in RO plant operation. However, FDI of RO plant sensors is equally important in order to ensure optimal and safe plant operation. FDI of plant sensors requires accurate plant and sensor models which are typically unavailable or impractical to develop for RO plants/processes that involve complex processes (i.e., involving coupled fluid flow, mass transfer and energy transport) whose nature (and coupling) often cannot be predicted a priori (e.g., due to fouling and mineral scaling, and gradual deterioration of plant components including sensors). The above challenge can be overcome, for the purpose of FDI, using data-driven plant process models developed using machine learning (ML) techniques which have been shown to provide highly accurate description of RO plant operation [21-24]. The advantage of ML models of plant operation is that they can be used for plant process data validation and reconciliation and thus data correction and imputation to compensate for abnormal or missing (respectively) sensor readings [21, 25-26].

In the present work, two approaches for RO plant sensor fault detection and isolation (FDI), as well as sensor data imputation (SenDI) were assessed based on the application of self-

organizing maps (SOM) and support-vector regression (SVR). The SVR and SOM machine learning methods to sensor FDI (termed here SFDI) are fundamentally different in their approach, the latter being a classification-based method that provides a visual portray of overall plant and sensor behavior, while the former is based on providing a quantitative data-driven model for relating output to input variables. The use of SOM for FDI has been proposed for various applications including, for example, vehicle cooling systems [27], aircraft engines [28], induction motors and electrical machines [29-30], power transformers [31-32] and anaesthesia systems [33]. Although SOM have been used to analyze the operation of RO systems [21-34], SOM analysis has not been previously proposed as the basis for FDI in RO plants. Accordingly, in the present work, FDI and SenDI approaches are developed, using the SOM and SVR machine-learning methods, and demonstrated based on operational data for a small laboratory spiral-wound RO pilot plant.

2. Experimental procedure

2.1 Feed solution and materials

Aqueous salt feed solutions were prepared using analytical grade sodium chloride (Fisher Scientific, ACS grade, Pittsburgh, Pennsylvania) in deionized (DI) water. The spiral-wound RO membranes utilized in pilot RO system (Dow Filmtec XLE-2540, The Dow Chemical Company, Midland, Michigan) were 2.5 inch (0.0635 m) outer diameter, 40 inch (1.02 m) long elements with per element surface area of 2.6 m^2. A single membrane element had a manufacturer reported permeate flow rate of 3.2 m^3/day, and salt rejection of 99%, determined at a pressure of 6.9 bar for a 500 mg/L NaCl feed solution.

2.2 RO system

Data for model development were generated using the UCLA spiral-wound Mini-Mobile-Modular (M3) pilot RO desalination system shown schematically in **Fig. 1** [3, 35-36]. The M3 system was designed for permeate water production capacity up to 1.2 m^3/h (7,560 gallons/day) for brackish water (5,000 mg/L TDS) operating at 75% recovery and up to 0.64 m^3/h (4,058 gallons/day) for seawater desalination (at recovery of 40%) using up to 18

spiral-wound elements in various configurations. In the present study, a configuration of six elements (one per pressure vessel) in series was utilized with the system operating in a total recycle mode with permeate and concentrate streams returned to the feed tank. Briefly, the M3 RO plant consisted of a 450 liters feed tank with two low-pressure feed pumps (Model JM3460-SRM, Sea Recovery, Carson, CA) pumping the RO feed through a series of cartridge microfilters (5 µm, 0.45 µm and 0.2 µm; 08P GIANT, pleated 177 polypropylene filter cartridges, Keystone Filter, Hatfield, PA). The RO plant was operated such that filtered feed was fed to the RO membranes via two-high pressure pumps (Danfoss Model CM 3559, 3HP, 3450RPM, Baldor Reliance Motor, Sea Recovery Corp. Carson, California) operating in parallel and controlled by variable frequency drives (VFDs) (Model FM50, TECO Fluxmaster, Round Rock, Texas). The retentate flow rate and pressure in the RO unit were set by a model-based controller [35] that adjusted both an electrically actuated needle valve (valve V-1) (model VA8V-7-0-10, ETI Systems, Carlsbad, California) on the retentate stream of the M3 RO system and the pump VFD. In order to maintain the RO feed temperature, a heat exchanger (Model BP 410-030 Refrigerant heat exchanger, ITT Industries) was installed on the retentate side of the RO system. Permeate and retentate streams were monitored via in-line conductivity sensors, conductivity/resistivity sensor electronics (Signet 2839 to 2842 and Signet 2850, George Fischer Signet, Inc. El Monte, California) and pH sensor (DryLoc pH electrodes 2775, George Fischer Signet, Inc. El Monte, California). The M3 plant was equipped with a centralized data acquisition system that received all sensor outputs (0-5 V, 0-10 V, 4-20 mA) and converted the signals to process variable values.

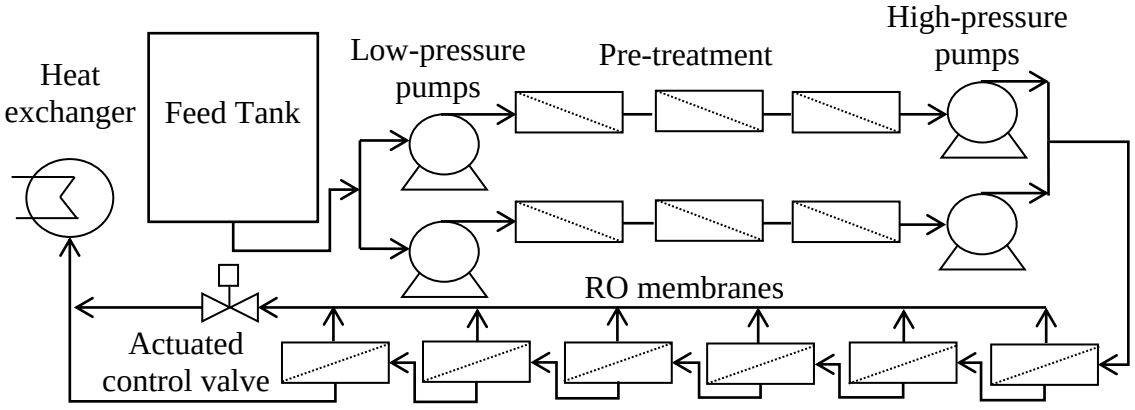

Figure 1. Configuration of the Spiral-wound RO pilot plant.

2.3 Experimental procedure

RO desalting experiments covered the range of operating conditions that were feasible by the operability limit of the system for the specific feed salinities. The experiments covered feed pressure and feed flow rate ranges of 7.9-24.8 bar (or 115-360 psi), and 0.26-0.68 m^3/h (or 1-3 gpm), respectively. The RO plant control system was programmed to autonomously step through a range of feed flow rates and transmembrane pressures, whereby each experiment was carried out until the attainment of steady state. Feed flow rate and pressure were varied by the system controller which provided the necessary adjustments of the high-pressure pumps VFD and actuated valve settings (**Fig. 1**). The data were logged into both the system embedded computer and a remote network database. Data could be logged at a frequency range as high as 1 kHz, although for the present study a frequency of a 1 Hz was deemed sufficient.

3. Model development

3.1 Data pre-processing

A total number of 81 different operational states, over the entire operational domain of the M3 RO plant (**Section 2.2**), were generated by inducing pressure and flow rate step changes of up to 15% and 25% respectively, with operational parameters recorded at a frequency of 1 Hz. The recorded raw data included natural plant fluctuations (i.e., due to operation of pumps and valves) as well as noise from normal operation of sensors, actuators and system pumps. Data were recorded from a total of eight different sensors (**Table 1**) and the average standard deviation (*ASTD*) for each measured sensor variable was determined from **Eq. 1**.

$$ASTD = \frac{1}{N}\sum_{i=1}^{N}\frac{STD_i}{y_{i,ave}} \times 100 \quad (1)$$

where *STD* is the standard deviation for the given variable within a steady state (for the set of operating conditions) trace *i*, $y_{i,ave}$ is the average variable value for the given trace, and *N* is the number of the experimental steady state traces.

Table 1. Range of steady-state plant operating parameters covered in the study.

Sensor Number	Variable	Range	Steady State STD[(a)]
1	Feed Flow rate (m^3/h)	0.26 - 0.68	1.21 %
2	Feed Conductivity (µS)	9,842 - 10,766	0.49 %
3	Feed Pressure (bar)	7.90 - 24.82	0.38 %
4	Permeate Flow rate (m^3/h)	0.06 - 0.28	0.94 %
5	Permeate Conductivity (µS)	629 - 1,823	0.55 %
6	Retentate Flow rate (m^3/h)	0.11 - 0.56	1.28 %
7	Retentate Conductivity (µS)	11,539 - 20,411	0.33 %
8	Retentate Pressure (bar)	3.59 - 10.75	0.98 %

(a) The steady state STD represents the average percent standard deviation of the readings for a given sensor, at steady state operation, relative to its average steady state value, based on all the steady state traces.

Data-driven models (i.e., SVR and SOM, see **Sections 3.2** and **3.4**, respectively) of the state-of-the-plant (i.e., steady state operation) were developed using steady-state data traces of 30 seconds, for the different steady state operational states, at 1 Hz sampling frequency. The steady-state period was established as that for which the measured process variables did not vary with time by more than 3% with respect to the time-averaged values. It is noted that the above short (30 s) traces at 1 Hz sampling frequency were sufficient for model development as longer operational data traces and higher sampling frequencies did not provide significant model improvements but did increased the computational burden.

The SVR models were developed with the sensor data (**Table 1**) that were normalized (for each sensor) in the range of [0, 1] using min-max normalization:

$$y_i^{(n)} = \frac{y_i - \min(y)}{\max(y) - \min(y)} \tag{2}$$

where $y_i^{(n)}$ is the normalized value of the experimental data y_i and *min(y)* and *max(y)* are the minimum and maximum variable values in the data set. Subsequently, for the purpose of SVR model building, the data were divided into two sets, one for model training and the other for model testing. The training data set was selected to cover the entire plant operational domain (i.e., applicability domain; [37]), by assuring that the maximum and minimum values of all variables were included in the training set. Data for model testing were selected randomly from the remaining data (i.e., not including the training set) but within the applicability domain to avoid extrapolation during predictions. In all cases, complete steady-state sequences were selected for the training and test sets.

Data for SOM generation were normalized to unit variance, in order to facilitate clustering of the different variables at the same scale (i.e., same mean and variance), such that:

$$y_i^{(n)} = \frac{y_i - \mu}{\sigma} \tag{3}$$

where $y_i^{(n)}$ is the normalized value of the experimental variable (y_i) and μ and σ are the mean and standard deviation of the variable values in the data set. It is noted that, in contrast to SVR models which are based on supervised learning, SOM is a neural network method based on unsupervised learning [38] which utilizes the entire data set.

In order to test the ability of both the SVR and SOM approaches to identify abnormal sensor behavior, anomalous sensor readings were generated by introducing perturbations relative to the real plant data (i.e., deviations). Synthetic faulty sensor behavior was generated by multiplying a trace of real plant sensor data by a perturbation vector of the same length as the sensor data trace, while the remaining sensor readings were kept at their real values (i.e., only one sensor was perturbed at a time). The generation of perturbed traces was repeated for each of the eight sensors, such that eight sets of faulty data (i.e., one for each of the eight sensors) were generated with each containing only one faulty sensor data. Sensor reading (SR) deviations that ranged from -50% to +50% relative to the real sensor readings in 10% intervals were then generated. SR deviations below ±4% were not assessed since the observed variability of sensor reading during steady state operation was up to about ±3%. Two different types of abnormal traces were generated: (a) a constant sensor drift (i.e., all values of the perturbation vector were the same for a specific deviation), (b) an initial progressive sensor drift that stabilizes at a specific SR deviation. For the second type of abnormal sensor behavior, the first one third of the sensor readings in the perturbation vector were set to either increase or decrease linearly until a preset percentage SR deviation (from the expected plant reading under normal plant operation) was attained, after which each successive sensor reading was taken to deviate by the same multiplier. It is important to recognize that within the framework of fault detection and isolation, when a given sensor reading is outside its nominal range of operation (i.e., above or below the range of operability as specified by the manufacturer) this sensor would be immediately identified as being faulty. However, sensor reading that are within the nominal

range could still be abnormal and thus one has to carry out a detailed fault detection and isolation as presented in the SVR and SOM approaches described in **Sections 3.2** and **3.3**, respectively.

3.2 Support vector regression (SVR) models

Steady-state models (**Sections 3.3**) were developed using the Support Vector Regression (SVR) algorithm [37, 39] using model training and test data sets (**Section 3.1**). Briefly, given a vector x of the RO process variables (e.g., feed pressure, feed flow rate, feed conductivity, retentate pressure, retentate flow rate, retentate conductivity, and permeate flow rate), the goal of SVR is to find a function $f(x)$ that has at most ε deviation with respect to the actual values of the target RO process variable y (e.g., permeate conductivity) and at the same time is as flat as possible (**Fig. 2**). SVR can be formulated as a convex optimization problem where a set of coefficients w for the regression model are computed in such a way that the flatness and accuracy of $f(x)$ are maximized. Since it is not always possible to keep the error within the margin ε for all the available data points, a pair of slack variables ξ and ξ^* must be introduced within the SVR formulation to cope with otherwise infeasible optimization constraints (**Fig. 2**). In most cases, the optimization problem can be solved more readily by projecting x (i.e., the vector of the input RO process variables) onto a higher-dimensional space where linear regression models can be developed for the target RO process variable of interest. The functions used to perform the above linear to non-linear mapping are known as kernel functions. Finally, the non-linear models that relate input and target variables are obtained by mapping the data back to the original (i.e., non-linear) space of the RO process variables. Only a subset of the training data points, representing the overall data behavior (i.e. support vectors), were used in the SVR formulation to generate the regression model (**Fig. 2**).

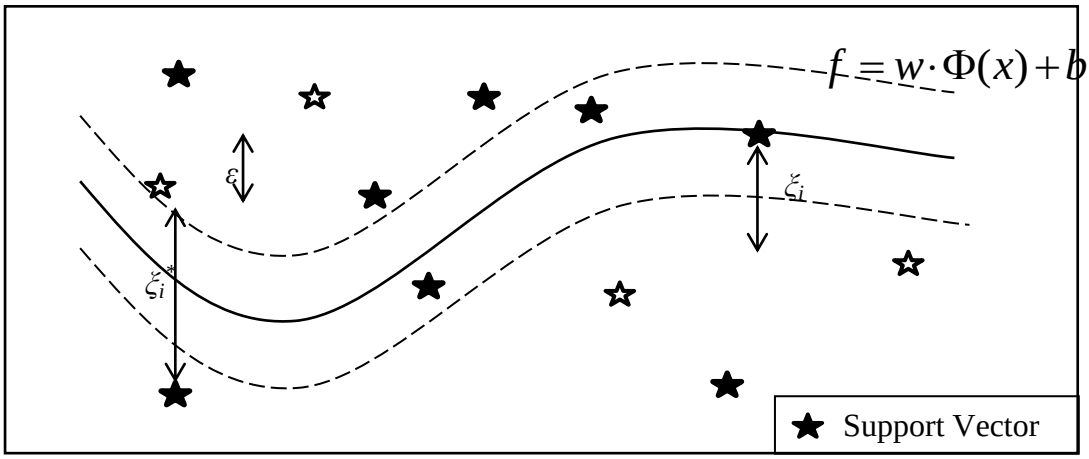

Figure 2. Support vector regression structural parameters. Regression function *f* supported on the most representative vectors of information (support vectors), *w* the normal vector to the hyperplane generated, the kernel function Φ, the vector of biases *b*, slack variables ξ and radius of the insensitive tube ε.

In the current work, the Radial Basis Function (RBF) was selected as the kernel function since it is suitable for systems of highly non-linear behavior [37]. The key parameter characterizing this kernel is the width of the Gaussian, σ_g; it determines the area of influence of the support vectors over the data space and here its optimal value was determined via a grid search. The SVR based models were developed in MATLAB using the LS-SVMlab1.7 package [40-41]. This SVR implementation utilizes a regularization parameter γ, which controls the tradeoff between the flatness (or smoothness) of the models and their accuracy, and the optimal parameter value was also determined via a grid search. The optimal values of γ and σ_g obtained via grid optimization for each model are summarized in **Table 2**.

3.3 Spiral-wound RO plant sensor fault detection and isolation (SFDI) scheme based on support vector regression (SVR) models

3.3.1 State-of-the-plant SVR models

The SFDI-SVR approach for detection of abnormal sensor behavior (**Fig. 3**) was based on assessing the deviation (i.e., quantified as residuals) between actual sensor readings and those expected for normal plant response as predicted from the SVR plant models. The use of SVR RO plant model was chosen given their previously demosntrated accurate performance for describing spiral-wound RO plant operation[21]. In the SFDI-SVR approach,

an individual state-of-the-plant (STP) SVR model was first developed for 7 of the plant process sensors excluding the feed conductivity ($cond_{feed}$) (**Table 1**). Although the total number of sensors in the plant is 8, the feed conductivity was considered as a variable independent of all other process variables which was not subject to process fluctuations. Feed conductivity sensor readings, however, were included as inputs to develop the SVR models for the remaining 7 sensors. Accordingly, SVR models were developed for each of the seven sensors (excluding the feed conductivity sensor),

$$y_s^* = f\left(y_j\right);\ \ j = 1,...,m+1;\ \ j \neq s; \tag{4}$$

where y_s^* is the predicted reading for sensor *s*, *j* designates the specific plant sensor, and *m* designates the seven sensors (i.e., total number of sensors excluding the feed conductivity sensor). For example, the SVR model for the permeate conductivity sensor (sensor 5 in **Table 1**) was based on input data from the remaining sensors listed in **Table 1** (i.e., 1-4 and 6-8, where $j \neq s$). For SVR model development, the steady state data traces (for the range of operating conditions listed in **Table 1**) were divided into training (62% of the dataset, 50 traces) and test (38% of the overall dataset, 31 traces) sets.

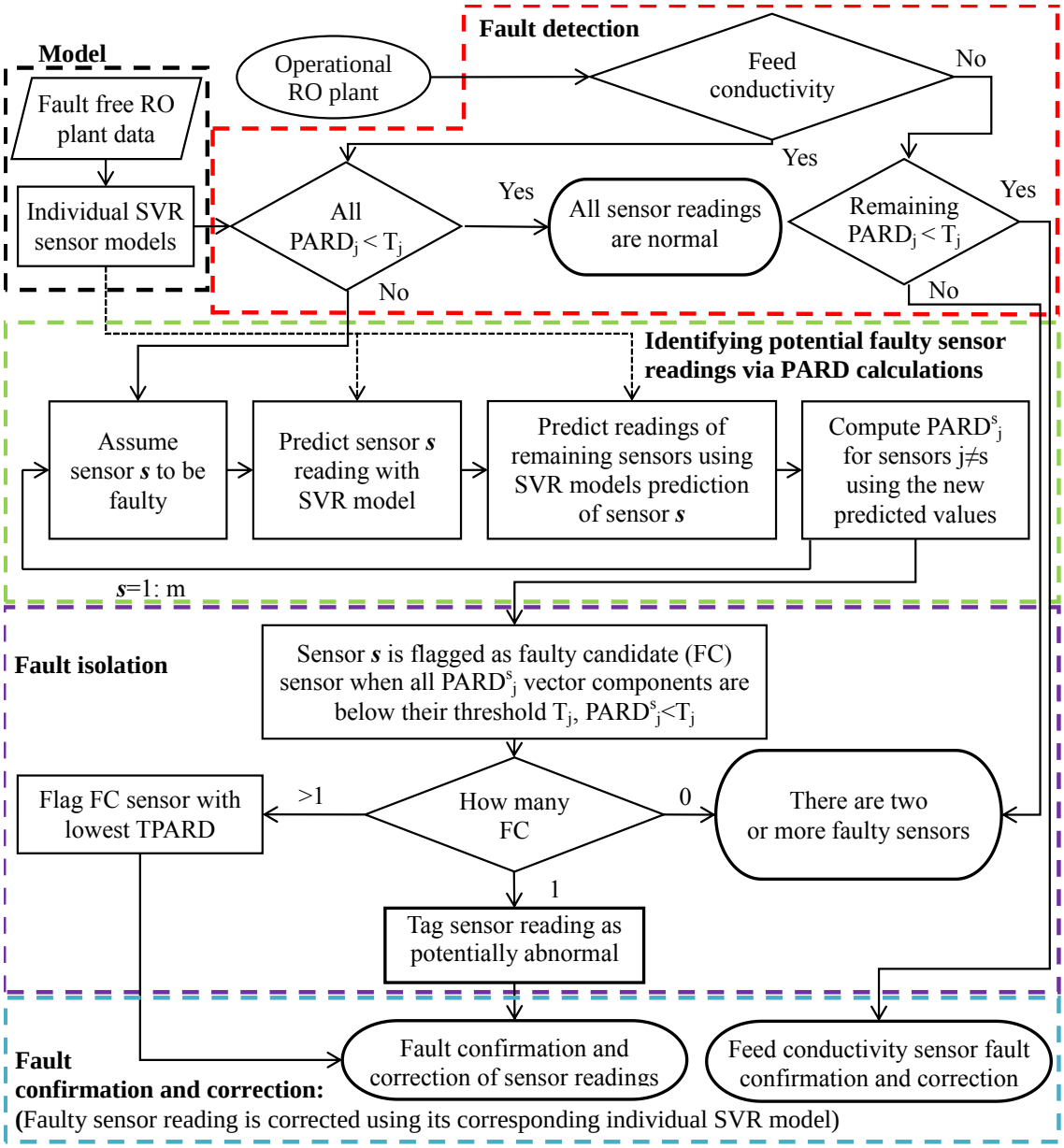

Figure 3. Support vector regression model-based algorithm for detection and isolation of abnormal sensor operation. The parameters *m*, *PARD*, *TPARD* and T refer to the number of plant sensors (not including the feed conductivity sensor), the percent absolute relative difference, total percent absolute relative difference between sensor reading and expected values and sensor threshold, respectively. $PARD^s_j$ contains the *PARD* of all sensors except for the *FC* sensor under consideration (i.e., $j \neq s$) and the feed conductivity sensor ($cond_{feed}$). Sensor *s* is identified as *FC* sensor when each $PARD^s_j$ of the remaining sensors is below its corresponding threshold T_j. If only one *FC* sensor is identified then it is directly flagged as being faulty. Note: If *FC* sensors are not encountered, this could suggest that more than one sensor may be faulty. If more than a single *FC* sensor is suspected, then the *FC* sensor with predictions closer to the experimental values (minimum *TPARD*) is selected as faulty.

Performance of the different SVR sensor models (with the optimal γ and σ_g values listed in **Table 2**) was quantified using the linear r^2 correlation coefficient (i.e., between the predicted, y_i^*, and experimental, y_i, variable values), and the average absolute (AAE) and percent average absolute relative (AARE) errors given as:

$$AAE_s = \frac{1}{n}\sum_{i=1}^{n}\left|y_{i,s}^* - y_{i,s}\right| \tag{5}$$

$$AARE_s = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_{i,s}^* - y_{i,s}}{y_{i,s}}\right| \times 100 \tag{6}$$

where n is the total number of data points for the dataset (training or test) and s is the sensor under consideration.

3.3.2 Fault detection and isolation with SFDI-SVR

It is important to note that fault detection and isolation for the conductivity sensor ($cond_{feed}$) sensor was based on the maximum deviation of sensor reading, from its expected value for the specific feed solution, over the course of each experiment. Therefore, feed conductivity sensor readings were assessed as the first step in the process of fault detection and isolation (**Fig. 3**). A fault in the $cond_{feed}$ sensor was inferred when its value differed more than 3.5% from the expected initially determined (preset). Once $cond_{feed}$ was confirmed to be correct, the fault detection and isolation strategy for the remaining sensors was carried out. Otherwise when the conductivity sensor ($cond_{feed}$) was found to be faulty, it was corrected given knowledge of the preset feed conductivity (**Section 3.3.3**) and the remaining sensors were checked for additional faults. Sensor fault detection and isolation (i.e., SFDI) strategy for the remaining seven sensors followed the basic structure of the so-called observer-based techniques[7]. Accordingly, the expected readings from each of the sensors are predicted by their respective SVR model and compared to their actual (experimental) readings as quantified by the percent absolute relative difference (PARD),

$$PARD_j = \frac{\left|y_j^* - y_j\right|}{y_j} \times 100 \tag{7}$$

where y_j^* and y_j are the predicted and experimental values of the sensor under consideration j. Sensor behavior is considered normal when $PARD_j$ is below a prescribed $PARD_j$

threshold, T_j, and is otherwise considered abnormal. In this study the sensor thresholds represent the maximum absolute sensor deviation, with respect to normal plant operation, which is deemed as acceptable (or allowable). Since the identity of the faulty sensor is not known a priori, each of the *m* plant sensors (i.e., total number of sensors excluding the feed conductivity sensor) are checked sequentially assuming that each one (in turn) could be a faulty sensor. Accordingly, in a sequential order, the expected reading, y_s^*, is predicted for each given sensor (*s*) of the plant's *m-1* sensors using the sensor's SVR model. It is noted that the SVR model for a given sensor *s* is developed based on the operational (experimental) readings of the other *m-1* sensors plus the feed conductivity sensor which are used as the SVR model input for the given sensor. For each sensor *s*, the $PARD^s_j$ for each of the remaining *m-1* sensors is then calculated and if the $PARD^s_j$ of all these *m-1* sensors are below the threshold T_j, then sensor *s* is flagged as faulty candidate (*FC*) sensor. The above isolation sequence is repeated for all the sensors. It is stressed, however, that SFDI becomes challenging when sensor deviations are small. Specifically, when sensor deviations are close to the *PARD* threshold, multiple faulty sensor candidates could be erroneously flagged as being potentially faulty. Also, if multiple sensors are corrected this could lead to erroneously identifying the remainder of the sensors as normal (i.e., accomplishing $PARD^s_j < T_j$, and thus identifying only one fault). In order to avoid the above dilemma, for each *FC* sensor the total percent absolute relative difference (*TPARD*) (**Eq. 8**) is evaluated as the sum of the *PARD* of the remaining sensors (i.e., the *PARD* for the FC sensor is not included in TPARD since its measured value is flagged as being potentially faulty). Note that the *FC* sensor reading needs to be predicted in order to use its value to predict the remaining *m-1* sensors by using the SVR model (**Eq. 4**),

$$TPARD_{FC} = \sum_{\substack{j=1 \\ j \neq FC}}^{m} PARD_j \tag{8}$$

where *j* designates the specific sensor (i.e., sensor number), *m* designates the seven plant sensors being assessed (i.e., total number of sensors excluding the conductivity sensor) and *FC* designate the sensor being assessed. The *FC* sensor of the minimum *TPARD* is identified as providing abnormal readings (i.e., faulty), while the remaining sensors are considered to be operating normally. The use of *TPARD* as the criterion for identifying the faulty sensor provides a relative measure of comparison of the deviation of predicted

compared to experimental sensor readings. Accordingly, the *FC* sensor with the smaller *TPARD* is more likely to be faulty (or providing abnormal readings).

3.3.3 Fault confirmation with SFDI-SVR

In order to confirm the presence of abnormal sensor readings, sensor faults are checked over the entire diagnostic period (**Fig. 3**). Abnormal measurements over a minimum number are needed before flagging a sensor as a fault candidate. This is required in order to avoid errant fault detection due to sensor measurement spikes or fluctuations which often occur in plant operation. In the present RO plant test case, a minimum of one third of the total trace measurements was set as being required in order to confirm a sensor as being flagged as faulty. Subsequently, the "faulty" sensor readings were reconstructed based on predictions from the sensor SVR model for the sensor using as input the readings from the other (*m-1*) sensors plus the feed conductivity sensor. When the feed conductivity sensor was identified as a potentially faulty sensor, the sensor reading was then replaced by the known present feed conductivity.

Although there is generally low likelihood of the simultaneous occurrence of multiple faulty sensors, the present approach can be extended to such cases. However, fault isolation would then require SVR sensor models based on, in addition to the feed conductivity sensor ($cond_{feed}$), *m-p* sensor readings for each modeled sensor, where *p* is the number of potential faulty sensors to be identified. Such an approach may be feasible provided that the number of faulty sensors is significantly less than the total number of sensors [6]. Notwithstanding, the present approach of dealing with incidences of a single faulty sensor occurrence should be of great practical applicability, particularly since it can also provide a warning regarding the potential occurrence of multiple faulty sensors.

3.4 Spiral-wound RO plant sensor fault detection and isolation (SFDI) scheme based on Self-Organizing Maps (SOM) analysis

3.4.1 SOM and RO Operational Domain

The Self-Organizing Map (SOM) clustering algorithm is used as an alternative approach for developing SFDI schemes (**Fig. 5**). The main advantage of SOM is that it can be used to characterize RO plant operational states in a manner that does not require individual model

for each plant sensor (**Section 3.3**). SOM also enables projection and visualization of multidimensional RO plant data into a 2D lattice (map) of SOM units arranged in a hexagonal geometry (i.e., each SOM unit has 6 neighboring units) [38, 42]. Each of the input data vector dimensions (i.e., 8 plant sensors of the present plant) is represented by a component plane (**Fig. 4**). The aggregation of all the component planes constitutes the complete SOM whose centroids contain information regarding all the data vector components (i.e., dimensions). In this approach, the identification of specific plant operation regimes is based on similarity calculations (i.e., Euclidean distance) between vectors formed by process variables (i.e., sensor readings) and real vectors that are represented by the centroid in each SOM unit. The SOM unit which is most similar to a given plant operational data vector is known as the "best matching unit" or *bmu*. In the current application, the vector $\mathbf{y} = \left[Q_f, C_f, P_f, Q_p, C_p, Q_r, C_r, P_r \right]$ represents each state of operation of the RO plant. Similarly, the centroid of a given unit *i* is a real vector $\mathbf{CE_i} = \left[\mu_1, \mu_2, ..., \mu_m \right]^{\mathbf{T}} \in \Re^m$ where *m* is the dimension of the input data (i.e., *m* represents the number of plant sensors; 8 for the present plant). During SOM development unit centroids are iteratively adapted to preserve, over the SOM projection, the topological relationships (i.e., ordering and distances) of the original high-dimensional RO plant operational space. As a consequence, similar SOM units are located close to each other resulting in clusters of similar operational states. In the current study, the SOM was developed using the MATLAB SOM Toolbox package [42].

A SOM for the domain of normal plant operation is first developed based on fault-free data (**Fig. 4**). Accordingly, the constructed SOM consists of SOM units encoding information from the various plant variables over the complete range of feasible operating conditions. Thus, the data vectors are classified using the SOM developed from fault-free plant data. The distance (i.e., similarity) of a given data vector relative to the centroid of its *bmu* (in the SOM built using the fault-free data) provides an indication of the consistency and reliability of the sensor reading relative to fault-free operational data. In other words, correct sensor readings are expected to be similar to readings obtained from fault-free plant data for the same operational state.

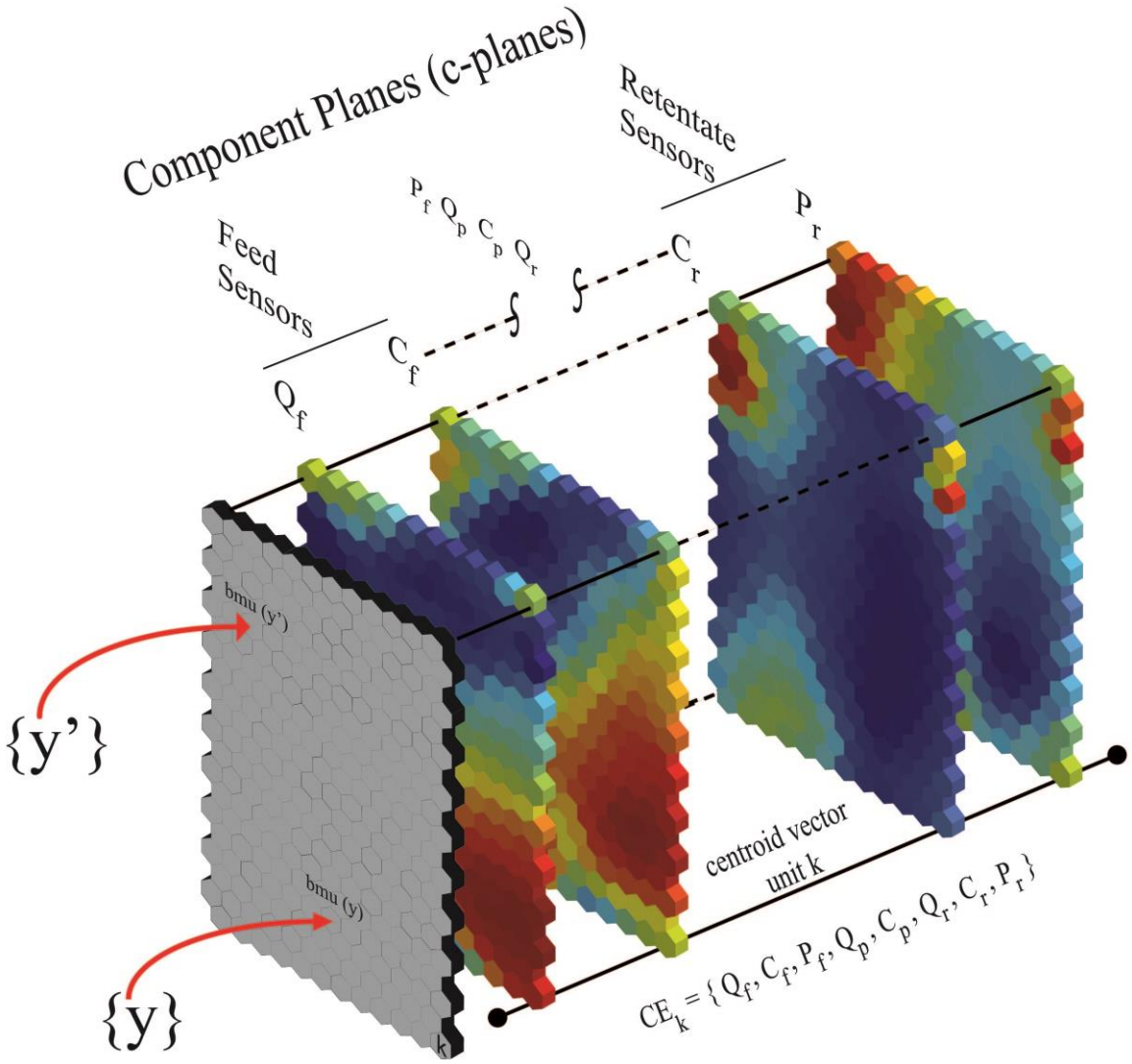

Figure 4. Each of the component planes (generated from its corresponding RO plant sensor data) are presented by a slice from the SOM. For the sake of clarity, the feed flow rate and conductivity, retentate pressure and retentate conductivity component planes are shown. Note that the overlaying SOM units contain information regarding a given plant operational state (e.g., SOM units along the solid line ended with circles, through the component planes, are for a similar plant operational domain).

3.4.2 Fault detection with SOM

In order to detect a sensor with faulty readings, an appropriate fault detection threshold was defined. As first step in the process, for each unit in the SOM, it is necessary to identify the data vector furthest away from the centroid (i.e., representing the average of all normal operational data vectors of the SOM unit) in terms of the Euclidean distance. The threshold was then defined as the absolute differences between the data vector components (i.e.,

readings of each sensor) and those of the centroid vector components, thereby yielding a vector containing a fault detection SOM (FDS) threshold (T_s) for each of the sensors. The FDS served to delimit the operational regime boundary captured by that SOM unit. It can be assumed that data vectors located beyond the FDS threshold have a significant deviation relative to the normal operation regime captured by the unit. Based on this criterion, the absolute difference (**AD**) vector between the components of a RO plant data vector **y** and the components of the centroid $\mathbf{CE}^{\mathbf{bmu(y)}}$ of its corresponding SOM *bmu* is first calculated,

$$AD_s^y = \left| CE_s^{bmu(y)} - y_s \right| \tag{9}$$

where y_s is the reading of sensor *s and* $CE_s^{bmu(y)}$ is the corresponding component of the *bmu* centroid. The data vector **y** is flagged as being faulty when any of its $AD_s^y > T_s^{bmu(y)}$, where $T_s^{bmu(y)}$ is the FDS threshold corresponding to sensor *s* for the given SOM *bmu*. Likewise, the data vector **y** is considered as not faulty when the AD_s^y values for all the eight plant sensors are smaller than their respective FDS thresholds ($T_s^{bmu(y)}$).

3.4.3 Fault isolation with SOM

Similar to the SVR-SFDI approach, in the SOM approach to fault isolation (i.e., identifying the faulty sensor once faulty data vector was identified) the feed conductivity sensor requires special attention since feed salinity is typically fixed by the feed source composition. Moreover, in the present work feed salinity (or conductivity) was fixed over the course of each experiment. Therefore, in the present approach, readings of the feed conductivity sensor were first checked to evaluate if those were faulty. A new data vector of *m* components was created excluding the feed conductivity sensor, and it was classified inside the SOM to find its *bmu* and calculate its **AD** vector. A fault in the feed conductivity sensor was identified when all the **AD** component values were below their corresponding FDS threshold ($T_s^{bmu(y)}$). Otherwise, the feed conductivity sensor was considered fault-free. In the process of fault isolation a new set of *m* reduced data vectors (**y**(m-1) where *m* is the number of plant sensors (excluding the feed conductivity sensor), each containing information from only *m-1* sensors (i.e., one sensor at a time is discarded in each of the new reduced data vectors along with the feed conductivity sensor), are generated from the original data vector. Therefore, with eight sensors in the present plant, 7 new reduced data vectors are created each containing 6 components. The *bmus* corresponding to these new

reduced data vectors are then identified, together with their corresponding **AD** vectors (**Eq. 9**). The fault isolation algorithm (**Fig. 5**) considers the following three different fault scenarios:

(a) *Only one of the seven reduced data vectors has all of its six **AD** vector components below their corresponding FDS thresholds*. In this case, the faulty sensor can be isolated and identified as the missing component in the above reduced data vector. The remaining reduced data vectors contain information from the faulty sensor and have at least one of their **AD** components above the FDS threshold;

(b) *The case where none of the seven reduced data vectors have all their **AD** components values below their FDS thresholds*. In the above situation, the set of SOM units most similar to the *bmu* (i.e., units located in the vicinity of the *bmu*) are also checked to ensure that there are no other neighboring SOM units (i.e., corresponding to similar operational states) with **AD** components below the FDS thresholds. In the present work, for all reduced data vectors, the maximum number of consecutive *bmus* that must be checked, to ensure that there are no other proximal units that may also have **AD** components below the FDS threshold, was set to nine (i.e., the six SOM units adjacent to the hexagonal *bmu* plus three additional SOM units with the shortest proximal distance from the *bmu*). The above conservative heuristic strategy (to reduce computational time) was employed in order to check if there are proximal SOM units next to the *bmu* that also satisfy the FDS threshold). When the first *bmu* that satisfies the FDS threshold (i.e., with **AD** values below the FDS threshold) is found, the confirmation and correction of the faulty sensor is then carried out as described in **Section 3.4.4**. However, when none of the first nine *bmus* satisfy the FDS threshold, fault isolation cannot be carried out since none of the reduced data vectors correspond to a normal plant operational state, which is an indication that there are likely to be two or more faulty sensors. Similar to the SFDI-SVR approach (**Section 3.3**), a new set of reduced data vectors, containing information for only *m-k* sensors, needs to be used in order to isolate *k* faulty sensors; and

(c) *The situation in which there is more than a single reduced data vector that has all its **AD** components below the FDS threshold, suggesting two or more faulty*

candidate (FC) sensors. For each FC sensor the total absolute difference TAD^{FC} (**Eq. 10**) is computed as the sum of AD components excluding the FC sensor.

$$TAD^{FC} = \sum_{\substack{s=1 \\ s \neq FC}}^{m} AD_s^y \tag{10}$$

In the above scenario, the reduced data vector (i.e., in which the FC sensor has been removed) with the lowest TAD^{FC} is identified as the data vector more likely to be fault-free, and therefore its discarded sensor component FC is inferred as faulty.

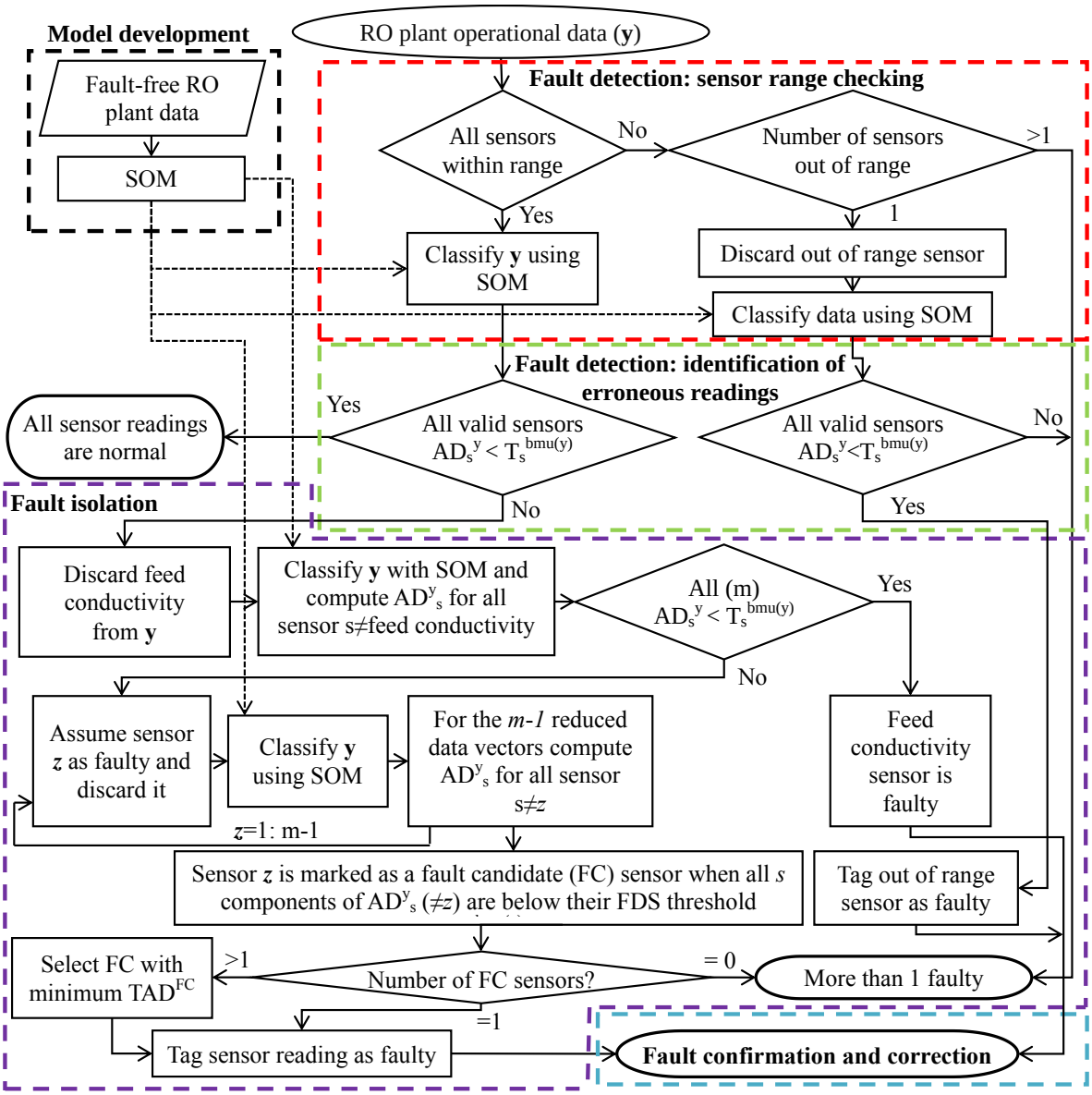

Figure 5. Detection and isolation algorithm for abnormal sensor operation based on self-Organizing Maps. The parameters *m*, *AD*, *TAD* and *T* refer to number of plant sensors (not including the feed conductivity sensor), absolute difference, total absolute difference and sensor FDS threshold, respectively. Note: The AD^{y}_{s} contains the *AD* of all sensors except the *FC* sensor under consideration (i.e., $s \neq z$) or except for the feed conductivity sensor ($cond_{feed}$) (i.e., $s \neq cond_{feed}$). A sensor *s* is identified as *FC* sensor when each AD^{y}_{s} of the remaining sensors is below its corresponding FDS threshold T_j. If only one *FC* sensor is identified then it is directly flagged as being faulty. Note: If *FC* sensors are not encountered, this could suggest that more than one sensor may be faulty. If more than a single *FC* sensor is suspected, then the *FC* sensor with predictions closer to the experimental values (minimum *TAD*) is selected as faulty.

3.4.4 Fault confirmation and correction.

Similar to the SFDI-SVR (**Section 3.3**), in the SFDI-SOM one also needs to ensure that the analysis is not biased by typical plant fluctuations that may be interpreted as produced by faulty sensor readings. Therefore, a minimum percentage of the diagnostic trace period is required to confirm abnormal sensor behavior and thus the approach was evaluated up to one third of the racer period (at 1 kHz data acquisition frequency; **Section 3.1**). Once a sensor fault is confirmed, the faulty sensor reading is corrected by replacing it with the corresponding component value in its *bmu* centroid.

4. Results and discussion

4.1. Performance of SFDI-SVR

State-of-the-plant (STP) models (**Section 3.3**) were first developed in order to provide predictions for each one of the sensors of the plant. Performances of STP models applied to the test data set are provided in **Table 2**. Models predicted feed, retentate and permeate flow rates with average absolute relative errors (AARE) of $2.3 \cdot 10^{-2}$%, $3.2 \cdot 10^{-2}$% and $2.3 \cdot 10^{-2}$% respectively. Similarly, feed, retentate and permeate conductivities were predicted with AARE levels of 0.33%, 0.32 % and 0.70%. Feed and retentate pressures were predicted with AARE levels 0.65% and 0.82% respectively. The greater accuracy of the flow rate compared to the pressure and conductivity sensor models. It is noted that the linear correlation coefficient (r^2) (for the predicted versus measured values) was in all cases ≥ 0.994. As discussed in **Section 3.3.1**, a model for the feed conductivity sensor was not implemented since the feed conductivity is an independent variable not affected by the remainder of the process variables. In fact, SVR model for this sensor (in terms of the

readings from the other seven sensors) yielded a model with a correlation coefficient of only 0.76. This was not surprising since the average deviation of the feed sensor relative to the steady state average was only about 0.49% while the stated manufacturer sensor performance was rated within an error of ±2%.

Table 2. Performance of SVR based models for predicting sensor values of RO plant operating at steady state conditions [(a)].

Predicted Variable	γ	σ_g	AAE	AARE (%)	r^2
Feed flow rate (m^3/h)	10,000	20	$1.1 \cdot 10^{-4}$	$2.3 \cdot 10^{-2}$	1
Feed pressure (bar)	200	20	$9.4 \cdot 10^{-2}$	0.65	0.994
Retentate conductivity (μS)	1,000	20	42.50	0.32	0.997
Retentate flow rate (m^3/h)	10,000	20	$1.2 \cdot 10^{-4}$	$3.2 \cdot 10^{-2}$	1
Retentate pressure (bar)	8,400	9	$5.0 \cdot 10^{-2}$	0.82	0.997
Permeate conductivity (μS)	200	20	8.23	0.70	0.999
Permeate flow rate (m^3/h)	10,000	14	$9.7 \cdot 10^{-5}$	$3.2 \cdot 10^{-3}$	1

(a) γ - regularization parameter; σ - width of the Gaussian; AAE and AARE are the average absolute and average absolute relative errors, respectively, and r^2 is the linear correlation coefficient.

The SFDI-SVR approach was evaluated for different sensor thresholds (**Table 3**). Steady state operation was considered established (**Section 3.1)** when the measured process variables did not vary with time by more than 3% with respect to the time-averaged values. Accordingly, the effect of the global sensor threshold was evaluated for sensor deviations over a range of 3.5% - 5% (i.e., slightly above the steady-state deviation of normal plant operation). Additionally, the SFDI-SVR approach was assessed with individual sensor thresholds for each sensor; these were based on each sensor's maximum deviation observed (relative to the average) during steady-state plant operation. Sensor thresholds for the retentate and permeate conductivity and feed pressure were set at 2%, feed flow rate at 2.5% and 3.5% for the retentate and permeate flow rate, retentate pressure and feed conductivity sensors (**Section 3.3.2)**. Using the above, performance of the SFDI-SVR

approach for sensor deviations (i.e., faults) between ±4% to ±50% and for sensor thresholds from 3.5% to 5%, as well as for individual sensor thresholds is shown in detail in **Table 3**.

Table 3. Performance of the SFDI-SVR for induced sensor deviations in the range of ±4% to ±50% and for both global and individual sensor thresholds [(a)].

	Individual Threshold		**Global Sensor Threshold**							
			3.5%		**4%**		**4.5%**		**5%**	
Dev. (%)	**FN (%)**	**FP (%)**	**FN (%)**	**FP (%)**	**FN (%)**	**FP (%)**	**FN (%)**	**FP (%)**	**FN (%)**	**FP (%)**
-50	0	0	0	0	0	0	0	0	0	0
-40	0	0	0	0	0	0	0	0	0	0
-30	0	0	0	0	0	0	0	0	0	0
-20	0	0	0	0	0	0	0	0	0	0
-10	0	0	0	0.40	0	0.40	0	0.40	0	0.40
-9	0	0.81	0	0.81	0	0.81	0	0.81	0	0.81
-8	0	0.81	0	0.81	0	0.81	0	0.81	0	0.81
-7	0	1.61	0	2.02	0	2.02	0	1.61	0	1.61
-6	0.4	2.42	0.4	3.23	0.4	3.63	0.4	3.63	0.4	3.23
-5	0.4	3.63	0.81	5.24	1.21	6.05	2.02	5.65	6.45	4.84
-4	1.61	7.26	4.03	6.85	8.47	7.26	34.27	5.65	48.39	5.24
+4	5.24	6.85	8.06	6.85	31.85	6.45	45.16	6.05	52.82	6.05
+5	2.42	4.44	2.82	4.84	5.24	4.84	8.06	4.84	34.27	4.84
+6	0.81	3.23	1.21	4.03	1.61	3.63	2.02	3.23	4.44	3.23
+7	0.4	2.42	0.4	2.42	0.81	2.42	0.81	2.42	1.21	2.02
+8	0.4	2.02	0.4	2.02	0.4	2.02	0.4	2.02	1.21	2.02
+9	0.4	0.81	0.4	0.40	0.4	0.81	0.4	0.81	0.4	0.81
+10	0	0.81	0	0.81	0	0.81	0	0.81	0.4	0.81
+20	0	0	0	0	0	0	0	0	0	0
+30	0	0	0	0	0	0	0	0	0	0
+40	0	0	0	0	0	0	0	0	0	0
+50	0	0	0	0	0	0	0	0	0	0

(a) FN and FP designate the percentages of false negatives and false positive of the total number of existent.

SFID-SVR performance (**Table 3**) was assessed in terms of false negative (FN) and false positive (FP) detections that to faulty sensor readings not identified as being faulty or fault-free sensor readings incorrectly identified as faulty. For both global and individual sensor fault detection thresholds, FN detections were zero (i.e., 100% detection of all faulty readings) for the above range of sensor thresholds (3.5%-5%) for up to sensor deviations of

±10%. FP detections occurred for $-10\% \leq \text{Sensor deviation} \leq 10\%$, with FP detection of 0.81% and 0.4% for sensor deviations of +10% and -10%, respectively, when using the global sensor thresholds. There were no FP detections for the above range of sensor deviations when using the individual sensor thresholds. When the SFDI-SVR algorithm (**Fig. 3**) was challenged with lower sensor deviations (i.e., ±4% to ±9%), the percentage of FPs and FNs increased as shown in **Fig. 6**. As expected, FN deviations increased with decreasing sensor deviations and increasing global sensor thresholds. On the other hand, FP detections remained essentially invariant with respect to global sensor thresholds but increased with decreasing sensor deviations. Overall, individual sensor thresholds demonstrated lower FN detections relative to the use of global sensor thresholds, while FP detections were similar for the two types of sensor thresholds.

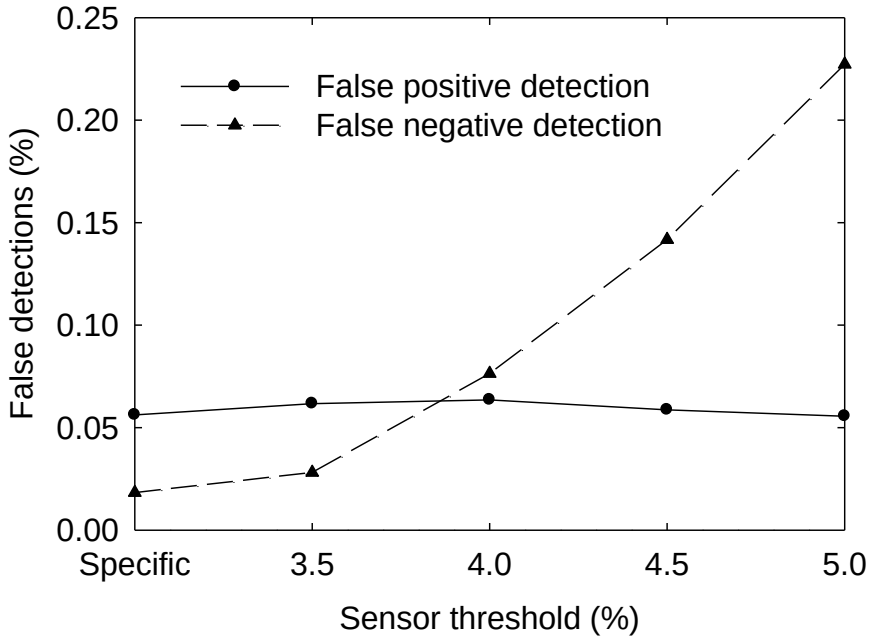

Figure 6. Total false positive and false negative detections for the plant eight sensors over the range of sensor deviations of ±4% to ±50%. The results are shown for both global and individual (first data points as specified in the sensor Threshold% axis) sensor thresholds.

In deploying the SFID-SVR it is desirable to minimize the number of FN detections from the viewpoint of plant safety and maintenance of process and equipment integrity. On the other hand, the occurrence of FP detections in a plant would trigger false alarms; these, however, can be minimized through a secondary process of using traditional decision tree analysis. It is noted that in the current study FP detections were encountered only for small sensor deviations. Clearly, small sensor deviations that are identified as either FP or FN detections are of lesser concern relative to those associated with large sensor deviations since the former are less likely to have significant impact on plant operation. As is evident from **Figure 6**, the number of FN detections increases (from 0.028%-0.23%) for higher sensor thresholds, while the number of FP detections decreases marginally (from 0.062%-0.056%) over the range of global sensor thresholds of 3.5% - 5%. Overall, it can be concluded that for the present RO pilot system, a greater allowance of sensor readings fluctuations, over a steady state operational period, can result in a higher rate of FN detections, whilst the number of FP detections is only slightly impacted. Therefore, the above observation suggests that individual sensor thresholds may be preferred when sensor performance accuracy data are available, while in the absence of the above global sensor thresholds can be utilized.

Table 4. Minimum sensor deviation that enabled first fault detection[(a)]

	Av. Det. Limit[(a)] (%)	**Accuracy of Sensor SVR Model, AARE[(b)] (%)**
Feed flow rate	3.45	0.02
Retentate flow rate	3.66	0.03
Permeate flow rate	5.56	0.09
Retentate Conductivity	3.73	0.32
Permeate Conductivity	3.73	0.70
Feed pressure	5.45	0.65
Retentate pressure	5.56	0.82

(a) Based on individual sensor thresholds; (b) Av. Det. Limit- minimum sensor deviations at which fault detection can be made (based on 372 data traces of sensor drift tests; **Section 3.1**); (b) AARE – average absolute relative error.

The SFDI-SVR method was also tested with increasing and decreasing sensor drifts (**Section 3.1**) in order to determine the minimum sensor deviations at which fault detection can be made (i.e., detection limit) with the detection performance summarized in **Table 4**. Accordingly, fault detection was evaluated for each one of the *m* plant sensors (not including the conductivity sensor). The minimum deviation (MD) for a given sensor that enabled fault detection, during a sensor drift (as described in **Section 3.1**), are provided **Table 4** representing the average (for each sensor) for 372 data traces of sensor drifts (i.e., for periods of faulty sensor operation). In order to check for consistency in the minimum sensor deviation at which fault detection can be made, this detection limit was compared for sensors measuring the same variable type. As shown in **Table 4**, the fault detection limits for the feed and retentate flow rates were similar (3.45% and 3.66% respectively), but higher (5.56%) for the permeate flow rate. The fault detection limits for the feed and retentate pressures were also similar (5.45% and 5.56%, respectively), and essentially identical for the retentate and permeate conductivity meters. As expected, the order of the fault detection limits correlated with the accuracy of the sensor SVR models (**Table 2**),

whereby, for similar type sensor measurements, SVR model of higher accuracy enabled lower fault detection limit.

Once a faulty sensor was identified, faulty data were corrected using the state-of-the-plant sensor SVR models (**Section 3.3.1).** Thus, data correction was within the same accuracy as that of the sensor SVR models (**Table 2**). As an illustration, the feed pressure sensor corrections, for two selected abnormal traces of sensor drifts (induced by increasing perturbations of up to ±5% relative to normal sensor operation) are shown in **Fig. 7**. It is important to recognize that sensor drifts can be mistaken for natural fluctuations of sensor readings when sensor deviations are small. However, the SFDI-SVR approach enabled detection of small sensor deviations (3.45%-5.56%) with a reasonable accuracy (**Tables 2** and **Table 5)**.

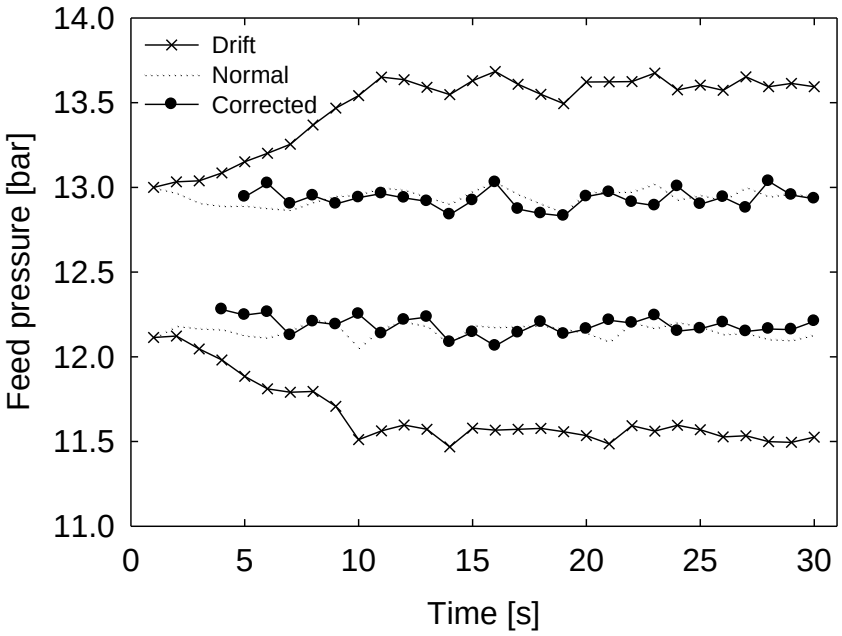

Figure 7. Corrections of feed pressure sensor for sensor readings with sensor drifts up to ±5% relative to expected normal operation values and using the individual sensor thresholds.

4.2. Performance of SFDI-SOM

In the first step of the SFDI-SOM approach, the fault-free RO plant data were utilized to construct a SOM that characterizes the normal plant operational states. This resulted in a 240 unit (20 rows and 12 columns) SOM with quantization and topographic errors of 0.397 and 0.214 respectively [42] for which a hexagonal grid was selected for the 2-D map representation. Although the shape of the complete SOM is toroidal, in the current work, for the sake of clarity, the SOM is presented as a series of 2-D component plane maps (**Figs 4 and 8).** In order to confirm detection of abnormal sensor behavior (**Section 3.4),** a minimum percentage of faulty readings (MPFR), of the total sensor readings in a prescribed period, was required to be identified as faulty. Accordingly, the performance of the SFDI-SOM method was evaluated, for MPFR (of the total data trace, **Section 3.1**) in the range of 13%-33% and sensor deviations (from normal operation) in the range of ±10% to ± 50% (**Table 5)**. In contrast to the SFDI-SVR approach, SFDI-SOM had a lower level of detection provision, whereby deviations below ±10% resulted in a significant number of FN detections; thus, the SFDI-SOM approach was only evaluated for sensor deviations of ±10% or above. While the above is a limitation, the SFDI-SOM approach has other advantages as discussed later in this section.

Table 5. Performance of SOM-based abnormal behavior detection and isolation tool for induced deviations from -50% to +50%. Percentage false negatives (FN) and false positives (FP) for the different points required to confirm faults.

	Percent of Faulty Sensor Data Required for Fault Confirmation					
	13%[(a)]		**20%[(a)]**		**33%[(a)]**	
Deviation (%)	**FN (%)**	**FP (%)**	**FN (%)**	**FP (%)**	**FN (%)**	**FP (%)**
+50	0	0	0	0	0	0
+40	0	0	0	0	0	0
+30	0.62	0.15	0.62	0	0.93	0
+20	1.7	0.31	1.7	0.15	1.85	0
+10	6.64	1.54	7.41	0.15	8.02	0
-10	3.24	4.17	3.55	1.54	4.17	0
-20	0.46	1.85	0.46	0	0.46	0
-30	0.15	1.85	0.15	0	0.15	0
-40	0	0.46	0	0	0	0
-50	0	0.46	0	0	0	0

(a) Percent of faulty data readings of the total data trace required for fault confirmation.

The number of undetected sensor faults (i.e., FN readings) was greater with increasing requirement of higher percentage of faulty data (of the total trace data) for fault confirmation, whilst the number of false positive (FP) detections decreased. For example, for sensor deviations of +20%, FN detections increased from 1.7% to 1.85% as the percentage of trace readings declared as faulty increased from 13% to 33%, while FP detections decreased from 0.31% to none for the same range. As expected, the number of FN and FP detections increased with smaller sensor deviations (from normal behavior), as it was more challenging for SFDI-SOM to identify such faults. For example, for sensor deviations of +10%, the percent FN and FP detections were as high as 6.64% and 1.54%, respectively (**Table 5**), based on the MPFR requirement of at least 13% for fault confirmation. As is evident from **Table 5**, one can tune the acceptance criterion of percentage of false negative relative to false positive detections by tuning the MPFR requirement for fault confirmation. Here we note again (as in **Section 4.1)**, that the

occurrence of FN detections is more problematic than FP detections; hence, one should establish fault confirmation criteria based on the acceptable balance between FN and FP detections.

Table 6. Accuracy of the corrections of the SFDI-SOM tool.

Corrected Variable	AAE[(b)]	AARE[(a)] (%)
Feed conductivity (μS)	20.31	0.19
Feed flow rate (m^3/h)	0.01	2.95
Feed pressure (bar)	0.42	2.75
Retentate conductivity (μS)	283	1.96
Retentate flow rate (m^3/h)	0.01	4.22
Retentate pressure (bar)	0.26	3.89
Permeate conductivity (μS)	40.81	3.37
Permeate flow rate (m^3/h)	0.02	3.96

Once the faulty sensors were detected and isolated, data reconciliation was carried out by replacing the faulty values with the values associated with the centroids of the *bmus* (see **Section 3.4.4**) corresponding to the reduced data vectors and discarding the faulty sensor data. The differences between the corrected values and the expected values (known from the experimental data) were calculated by means of the AAREs, being in all cases smaller than 5%. The accuracy of data correction (of faulty sensors) via the SFDI-SOM approach was in the range AARE of 0.19%-4.22% for the eight plant sensors (**Table 6**). The above level of accuracy for faulty data correction is somewhat lower than that which was achieved (0.02%-0.82% AARE) for the SFDI-SVR approach. However, the SFDI-SOM is simpler to use as it does not require individual sensor models but only a single SOM construction based on the fault-free plant operational data.

An attractive element for using SOM, which can make its use beneficial as either an add-on to SFDI-SVR or SFDI-SOM is in that SOM can provide a visual representation of sensor deviations from normal plant operation. For example, based on the historical plant data

used to construct the plant SOM and the desired operational domain, one can establish visually the location of a given plant operation state over the SOM. When the plant operates under normal conditions, the *bmus* representing plant data are located in a contiguous SOM area corresponding to normal plant operation. In contrast, when sensor readings are faulty, significant drifts of the *bmu* are observed. As an illustration, **Figure 8** shows one example of the effect of an increasing perturbation up to +50% on the feed flow rate sensor. Each subfigure represents the so-called component planes of the SOM, which are generated by plotting each individual component of the centroid of the SOM units. Accordingly, each component plane depicts the distribution of the sensor readings for each of the 8 plant sensors (i.e., retentate, feed and permeate flow rates, feed and permeate pressures, and retentate, permeate and feed conductivities) over the SOM grid. Note that each component plane corresponds to a slice of the SOM and therefore overlaying SOM units are all for the same plant operational state (see **Figure 4**). In the example of **Figure 8**, normal plant operation associated with SOM unit marked with "*" and the deviated operational state resulting from information provided by the faulty feed flow rate is depicted by black or white circles. **Figure 8** shows the *bmu* of the faulty data moving away (i.e., trajectory) from the location corresponding to the normal operation *bmu* (i.e., *bmu* of the vector with fault-free operational parameters). The drift in sensor readings is apparent in the example of **Figure 8**, where the *bmu* associated with the faulty data vector drifts, along the shown trajectory of increasing time (at 5 second intervals), further away from the *bmu* associated with the expected normal plant operation (marked with "*"). The above depiction of the state of the plant and sensor deviation can provide a rapid and visual aid for plant operators to monitor plant operational status and to detect abnormal plant behavior. The above visual approach does not identify the specific faulty sensor, but can provide an early warning

which could then proceed with the process of faulty sensor identification and isolation (by either SFDI-SVR or SFDI-SOM) as described in **Sections 3.3.2** and **3.4.3.**

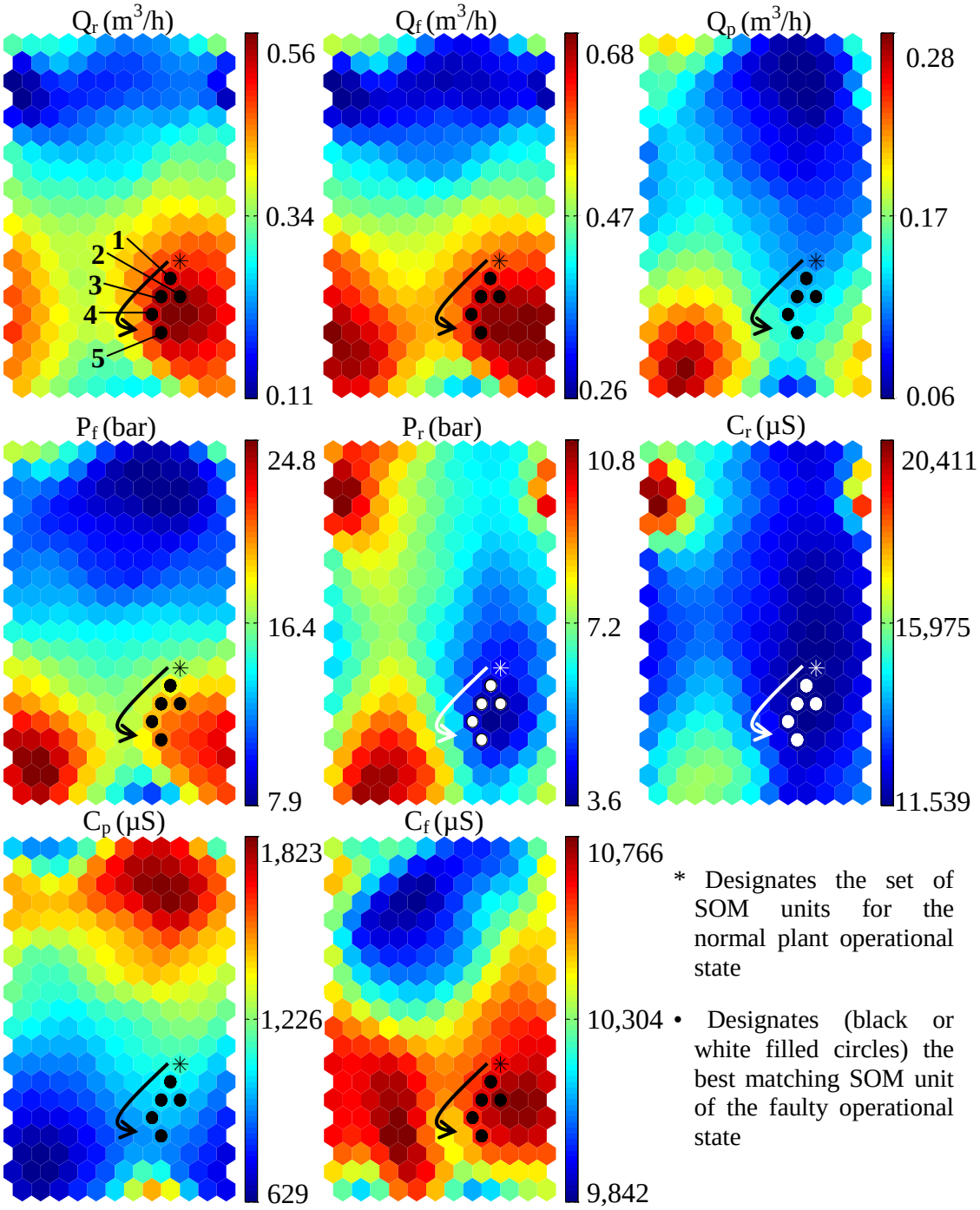

Figure 8. Plant behavior trajectory for a data trace with a drift in the feed flow rate sensor reading. The deviation from the nominal value increases linearly with time up to a deviation of +50%. Note: The trajectory of the best matching unit (*bmu*) for the faulty data vector with time is marked as 1-5 (at successive 5 s intervals).

5. Conclusions

Sensor fault detection and isolation (SFDI), as well as sensor data corrections approaches for spiral-wound RO plant operation were assessed based on the use of both support-vector regression (SVR) plant sensor models and self-organizing map (SOM) representation of plant operation. SFDI-SVR was based on data-driven models developed for individual RO plant sensors using the support vector regression model building approach. SFDI-SVR enabled fault detection without false negative detections for sensor deviations in the range of ±10% or greater when sensor thresholds (global or individual) were just below 5%. False positive detections were higher than false negative detections by up a factor of two and greater in some cases, particularly for increasing global sensor detection threshold and small sensor deviations (e.g., approaching the expected normal sensor fluctuations under normal steady state conditions). Overall, individual sensor thresholds provided lower false negative detection rate for all sensor deviations, but false positive detections appeared to be nearly invariant with respect to sensor threshold in the range of 3.5%-5%. Corrections of faulty sensor readings were within the accuracy level of the SVR sensor models (based on evaluation of sensor performance for normal plant operation). The SFDI-SOM's performance was inferior to SFDI-SVR, demonstrating no false negative detections only for sensor deviations in the range of ±40% or higher, with increased false negative detections of 3.24%-6.64% for sensor deviations in the range of ±10% with SOM minimum percentage of faulty readings (MPFR) of 13% (i.e., percent of sensor reading in a given monitoring trace being identified as faulty) for fault confirmation and somewhat higher (4.17%-8.02%) for MPFR of 33%. Corrections of the abnormal sensor readings with SFDI-SOM were with AARE lower than 5%. Although SFDI-SOM was of lower accuracy than

SFDI-SVR, its application is simpler, since it does not require developing a model for each of the plant sensors. Moreover, SOM provides a pictorial representation of plant operation and trajectory of faulty sensor behavior that could be beneficial to plant operators. Overall, both approaches were robust and able to handle noisy plant data. Clearly the choice of using either one or integration of both of the above approaches for RO plants would have to be made on the basis of acceptability of the rates of false negative relative to false positive fault detections and desirability for visual representation of plant operation as feasible via SOM.

Nomenclature

AAE	average absolute error
AARE	average absolute relative error
AD	absolute difference
ANN	artificial neural networks
b	bias vector
bmu	best matching unit
C_f	feed conductivity
C_p	permeate conductivity
C_r	retentate conductivity
CE	centroid
$cond_{feed}$	feed conductivity sensor reading
FDI	fault detection and isolation
FDS	fault detection SOM

Nomenclature	
FN	false negative
FP	false positive
FTC	fault tolerant control
f(x)	function obtained by SVR
m	number of sensors not including feed conductivity
n	number of experimental values
P_f	feed pressure
P_r	retentate pressure
PARD	percent absolute relative difference
PCA	principal component analysis
Q_f	feed flow rate
Q_p	permeate flow rate
Q_r	retentate flow rate
RO	reverse osmosis
r^2	correlation coefficient
s	sensor under consideration
SFDI	sensor fault detection and isolation
SOM	self-organizing-map
STDV	standard deviation
SVM	support vector machines
SVR	support vector regression
T	threshold

Nomenclature	
TAD	total absolute difference
TDS	total dissolved solids
TPARD	total percent absolute relative difference
V_i	average variable value for the given steady state
VFD	variable frequency drive
w	normal vector to hyperplane
x	vector of input RO parameters in the SVR model
y	vector of input parameters in the SOM
y_i	experimental value for a given variable
y_i^*	predicted value for a given variable
$y_i^{(n)}$	normalized value for a given variable
ε	tolerance of the SVR models
γ	regularization parameter
ξ, ξ^*	upper and lower slack variables
σ	variance / width of the Gaussian
Φ	kernel function

AUTHOR INFORMATION

Corresponding Author

*Tel.: +01 310-825-8766. Fax: +01 310-206-4107. Email: yoram@ucla.edu.

Notes: The authors declare no competing financial interest.

ACKNOWLEDGEMENTS

This work was supported in part by funding from the Catalan Government (2009 SGR 1529), the Spanish Ministry of Education (CTQ2009-14627), the California Department of Water Resources, the United States Environmental Protection Agency, the National Water Research Institute, the Office of Naval Research, and the UCLA Water Technology Research Center.

REFERENCES

[1] Jamal, K.; Khan, M. A.; Kamil, M., Mathematical modeling of reverse osmosis systems. *Desalination* **2004,** *160* (1), 29-42.
[2] Abbas, A., Model predictive control of a reverse osmosis desalination unit. *Desalination* **2006,** *194* (1–3), 268-280.
[3] Bartman, A. R.; Christofides, P. D.; Cohen, Y., Nonlinear Model-Based Control of an Experimental Reverse-Osmosis Water Desalination System. *Industrial & Engineering Chemistry Research* **2009,** *48* (13), 6126-6136.
[4] Bartman, A. R.; McFall, C. W.; Christofides, P. D.; Cohen, Y., Model-predictive control of feed flow reversal in a reverse osmosis desalination process. *Journal of Process Control* **2009,** *19* (3), 433-442.
[5] Venkatasubramanian, V.; Rengaswamy, R.; Yin, K.; Kavuri, S. N., A review of process fault detection and diagnosis: Part I: Quantitative model-based methods. *Computers & Chemical Engineering* **2003,** *27* (3), 293-311.
[6] Mhaskar, P.; Liu, J.; Christofides, P. D., *Fault-tolerant Process Control: Methods and Applications*. Springer: 2013.
[7] Patton, R. J.; Chen, J., Observer-based fault detection and isolation: Robustness and applications. *Control Engineering Practice* **1997,** *5* (5), 671-682.
[8] Isermann, R., Model-based fault-detection and diagnosis–status and applications. *Annual Reviews in control* **2005,** *29* (1), 71-85.
[9] McFall, C. W.; Christofides, P. D.; Cohen, Y.; Davis, J. F. In *Fault-tolerant control of a reverse osmosis desalination process*, Proceedings of 8th IFAC Symposium on Dynamics and Control of Process Systems, 2007; pp 163-168.
[10] Eski, I.; Erkaya, S.; Savas, S.; Yildirim, S., Fault detection on robot manipulators using artificial neural networks. *Robotics and Computer-Integrated Manufacturing* **2011,** *27* (1), 115-123.
[11] Chilin, D.; Liu, J.; Davis, J. F.; Christofides, P. D., Data‐based monitoring and reconfiguration of a distributed model predictive control system. *International Journal of Robust and Nonlinear Control* **2012,** *22* (1), 68-88.

[12] Jack, L.; Nandi, A., Fault detection using support vector machines and artificial neural networks, augmented by genetic algorithms. *Mechanical systems and signal processing* **2002,** *16* (2), 373-390.
[13] Samanta, B.; Al-Balushi, K.; Al-Araimi, S., Artificial neural networks and support vector machines with genetic algorithm for bearing fault detection. *Engineering Applications of Artificial Intelligence* **2003,** *16* (7), 657-665.
[14] Gambier, A.; Miksch, T.; Badreddin, E. In *A reverse osmosis laboratory plant for experimenting with fault-tolerant control*, American Control Conference, 2009. ACC'09., IEEE: 2009; pp 3775-3780.
[15] McFall, C. W.; Bartman, A.; Christofides, P. D.; Cohen, Y., Control and Monitoring of a High Recovery Reverse Osmosis Desalination Process. *Industrial & Engineering Chemistry Research* **2008,** *47* (17), 6698-6710.
[16] Garcia-Alvarez, D.; Fuente, M. J., A UPCA-based monitoring and fault detection approach for reverse osmosis desalination plants. *Desalination and Water Treatment* **2013**, 1-15.
[17] Guang-hong, Y.; Si-Ying, Z.; Lam, J.; Jianliang, W., Reliable control using redundant controllers. *Automatic Control, IEEE Transactions on automatic control* **1998,** *43* (11), 1588-1593.
[18] ŠIljak, D. D., Reliable control using multiple control systems. *International Journal of Control* **1980,** *31* (2), 303-329.
[19] *EcosimPro*, EA International: Madrid.
[20] Arteaga, F.; Ferrer, A., Framework for regression-based missing data imputation methods in on-line MSPC. *Journal of Chemometrics* **2005,** *19* (8), 439-447.
[21] Pascual, X.; Gu, H.; Bartman, A. R.; Zhu, A.; Rahardianto, A.; Giralt, J.; Rallo, R.; Christofides, P. D.; Cohen, Y., Data-driven models of steady state and transient operations of spiral-wound RO plant. *Desalination* **2013,** *316*, 154-161.
[22] Khayet, M.; Cojocaru, C.; Essalhi, M., Artificial neural network modeling and response surface methodology of desalination by reverse osmosis. *Journal of Membrane Science* **2011,** *368* (1–2), 202-214.
[23] Libotean, D.; Giralt, J.; Giralt, F.; Rallo, R.; Wolfe, T.; Cohen, Y., Neural network approach for modeling the performance of reverse osmosis membrane desalting. *Journal of Membrane Science* **2009,** *326* (2), 408-419.
[24] Abbas, A.; Al-Bastaki, N., Modeling of an RO water desalination unit using neural networks. *Chemical Engineering Journal* **2005,** *114* (1–3), 139-143.
[25] Narasimhan, S.; Jordache, C., 11 - Industrial Applications of Data Reconciliation and Gross Error Detection Technologies. In *Data Reconciliation and Gross Error Detection*, Gulf Professional Publishing: Burlington, 1999; pp 327-372.
[26] Batur, C.; Ling, Z.; Chien-Chung, C. In *Support vector machines for fault detection*, 2002, Proceedings of the 41st IEEE Conference on Decision and Control, 10-13 Dec. 2002; 2002; pp 1355-1356 vol.2.
[27] Svensson, M.; Byttner, S.; Rognvaldsson, T. In *Self-organizing maps for automatic fault detection in a vehicle cooling system*, Intelligent Systems, 2008. IS '08. 4th International IEEE Conference, 6-8 Sept. 2008; 2008; pp 24-8-24-12.

[28] Cottrell, M.; Gaubert, P.; Eloy, C.; François, D.; Hallaux, G.; Lacaille, J.; Verleysen, M., Fault Prediction in Aircraft Engines Using Self-Organizing Maps. In *Advances in Self-Organizing Maps*, Príncipe, J.; Miikkulainen, R., Eds. Springer Berlin Heidelberg: 2009; Vol. 5629, pp 37-44.
[29] Bossio, J. M.; De Angelo, C. H.; Bossio, G. R.; Garcia, G. O. In *Fault diagnosis on induction motors using Self-Organizing Maps*, 2010 9th IEEE/IAS International Conference on Industry Applications (INDUSCON), 8-10 Nov. 2010; 2010; pp 1-6.
[30] Fei, Z.; Tielin, S.; Tao, H. In *Fault diagnosis of motor bearing using self-organizing maps*, Electrical Machines and Systems, 2005. ICEMS 2005. Proceedings of the Eighth International Conference on, 29-29 Sept. 2005; 2005; pp 2411-2414 Vol. 3.
[31] Thang, K. F.; Aggarwal, R. K.; McGrail, A. J.; Esp, D. G., Analysis of power transformer dissolved gas data using the self-organizing map. *Power Delivery, IEEE Transactions on Power Delivery* **2003,** *18* (4), 1241-1248.
[32] Pengju, K.; Birtwhistle, D., Condition assessment of power transformer onload tap changers using wavelet analysis and self-organizing map: field evaluation*, IEEE Transactions on Power Delivery* **2003,** *18* (1), 78-84.
[33] Vapola, M.; Simula, O.; Kohonen, T.; Meriläinen, P., Representation and Identification of Fault Conditions of an Anaesthesia System by Means of the Self-Organizing Map. In *ICANN '94*, Marinaro, M.; Morasso, P., Eds. Springer London: 1994; pp 350-353.
[34] Libotean, D.; Giralt, J.; Rallo, R.; Cohen, Y.; Giralt, F.; Ridgway, H. F.; Rodriguez, G.; Phipps, D., Organic compounds passage through RO membranes. *Journal of Membrane Science* **2008,** *313* (1–2), 23-43.
[35] Bartman, A. R.; Zhu, A.; Christofides, P. D.; Cohen, Y., Minimizing energy consumption in reverse osmosis membrane desalination using optimization-based control. *Journal of Process Control* **2010,** *20* (10), 1261-1269.
[36] Gu, H.; Bartman, A. R.; Uchymiak, M.; Christofides, P. D.; Cohen, Y., Self-adaptive feed flow reversal operation of reverse osmosis desalination. *Desalination* **2013,** *308* , 63-72.
[37] Bishop, C. M., *Pattern Recognition and Machine Learning*. Springer Science+ Business Media, LLC: New York, NY, 2006.
[38] Kohonen, T., *Self-organizing maps*. Springer: 2001; Vol. 30.
[39] Vogt, M.; Kecman, V., Active-set methods for support vector machines. In *Support vector machines: Theory and applications*, Wang, L., Ed. Springer: Berlin, 2005; Vol. 177, pp 133-158.
[40] Vapnik, V. N., *The nature of statistical learning theory*. Springer-Verlag New York, Inc.: 1995; p 188.
[41] De Brabanter, K.; Karsmakers, P.; Ojeda, F.; Alzate, C.; De Brabanter, J.; Pelckmans, K.; De Moor, B.; Vandewalle, J.; Suykens, A. K. *LS-SVMlab Toolbox user's guide*; K.U. Leuven Leuven, Belgium, 2011.
[42] Alhoniemi, E.; Himberg, J.; Parhankangas, J.; Vesanto, J. *SOM Toolbox*, 2.0; 2000.

9. Conclusions

This thesis has presented two successful applications of data-driven models developed for a reverse osmosis desalination pilot plant. The Support Vector Regression (SVR) modeling approach for steady state operation of the reverse osmosis pilot plant M3 has performed better results than the commonly used first principle-based models. In the same way, the dynamic models have enabled the short-term prediction and simulation of the M3 plant behavior under non-steady state operation, with such a high accuracy that it makes the approach suitable for advanced reverse osmosis plant control algorithms, fault tolerant control and process optimization. In fact, SVR models have been proved to perform excellent results as part of a fault detection and isolation tool for the M3 plant. In addition, an alternative to SVR models which uses Self-Organizing Maps (SOM) has been also demonstrated for fault detection and isolation. The SOM-based tool has not reached the performance levels obtained by the SVR-based tool, although it has provided a useful visual tool for the rapid fault detection during plant operation. It is important to note that all the algorithms and tools developed in this study which are based on data-driven models are specific for the M3 plant, and must be applied only for the applicability domain covered in this study. In this respect, the applicability domain of the models can be extended at the same time as new plant data are acquired by implementing online model training.