UNIVERSITY OF CALIFORNIA

Los Angeles

Control and Monitoring of Reverse Osmosis Water Desalination

A dissertation submitted in partial satisfaction of the

requirements for the degree Doctor of Philosophy

in Chemical Engineering

by

Alex Robert Bartman

2011

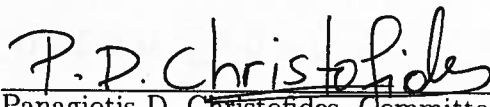The dissertation of Alex Robert Bartman is approved.

Tsu-Chin Tsao

Harold G. Monbouquette

Yoram Cohen, Committee Co-Chair

Panagiotis D. Christofides, Committee Co-Chair

University of California, Los Angeles

2011

ii

Contents

List of Figures

List of Tables

xxx

ACKNOWLEDGEMENTS

I would like to thank my family for their love and support, especially Mom, Dad and Emily. Without their encouragement, assistance, and allowing me to pursue what I enjoyed, attaining this degree would not have been possible. I would also like to show my gratitude to my Ph.D. faculty advisors, Panagiotis D. Christofides and Yoram Cohen, for their patience and time spent in helping lead me during the course of my time at UCLA. Their advice and ideas were instrumental in giving me the ability to adapt to the research environment and giving me consistent motivation through their passion for the work.

I would also like to thank my colleagues that worked closely with me on the group projects, including Charles McFall, Han Gu, Anditya Rahardianto, Larry Gao, Xavier Pascual Caro, and Robert Rallo, for their input and assistance during my time at UCLA. I very much appreciate Andi's tireless efforts to keep the group focused, as well as his guidance in the preparation of my dissertation. I am also indebted to a good friend and colleague in Larry Gao; without his work ethic and positive attitude, the major research undertakings of the CoM2RO project would not have been successful. I also owe many of my research accomplishments to the patience and guidance of Charles McFall who helped me build the basis for my work.

I am also grateful to have worked with a number of helpful and knowledgeable people in both groups, including Jinfeng Liu, Gangshi Hu, Ben Ohran, Patrick Sislian, Michael Nayhouse, Brian McCool, Eric Lyster, and Nancy Lin. I very much appreciate

the discussions with Eric about our computer code projects, as well as the games of cribbage and guitar sessions. I am also honored to have a wonderful friend and confidant in Nancy, who helped me in countless ways with her insight and was always willing to listen.

During my time in Los Angeles I was fortunate to meet and get to know a number of people who made the graduate school experience much more enjoyable. I would like to thank Jon Tesch, Kitty Cha, Diana Chien, James Dorman, Kevin Smith, and Sara Habib for being great friends, as well as my old friends who tolerated my hectic schedule and still put forth the effort to keep in touch with me, JD Rossman, Russell Derrick, Mackie Derrick, Tyler Butler, and Erin Waddell.

I must also thank the UCLA CBE administrative staff, especially Victoria Ybiernas and John Berger, for their immense assistance, as well as funding from the California Department of Water Resources and the Office of Naval Research for making the research work possible.

VITA

June 1, 1984   Born, Neenah, Wisconsin, USA

2005   Bachelor of Science, Chemical Engineering

                  Illinois Institute of Technology

                  Chicago, Illinois

2005   Graduate Student Researcher

                  Illinois Institute of Technology

2006   Visiting Graduate Student Researcher

                  University of Auckland, New Zealand

2006–2011   Research Assistant, Teaching Assistant, Associate, and Fellow

                  Department of Chemical and Biomolecular Engineering

                  University of California, Los Angeles

PUBLICATIONS AND PRESENTATIONS

1. A. R. Bartman, H. Gu, L. Gao, A. Zhu, J. Thompson, B. McCool, A. Rahardianto, P. D. Christofides, Y. Cohen, "Design, Operation and Control of Next-Generation Water Purification Systems Utilizing Reverse Osmosis Desalination: Commercial Scale System Design, Construction, and Field Testing," UCLA Chemical and Biomolecular Engineering Department Seminar Series, May 27, 2011, Los Angeles, CA.

2. A. R. Bartman, A. Zhu, H. Gu, L. Gao, J. Thompson, A. Rahardianto, P. D. Christofides, Y. Cohen, "Optimal Operation and Control of Reverse Osmosis Water Desalination," Southern California Nonlinear Control Workshop, May 13, 2011, Riverside, CA.

3. A. R. Bartman, R. Rallo, P. D. Christofides, Y. Cohen, "Real-Time Monitoring of Mineral Scaling for Reverse Osmosis Desalination," AWWA Membrane Technology Conference and Exposition, Mar. 30, 2011, Long Beach, CA.

4. A. R. Bartman, R. Rallo, P. D. Christofides, Y. Cohen, "Real-Time Monitoring of Mineral Scaling for Reverse Osmosis Desalination," UCLA Technology Forum, Mar. 1, 2011, Los Angeles, CA.

5. A. R. Bartman, E. Lyster, R. Rallo, P. D. Christofides, Y. Cohen. Mineral scale monitoring for reverse osmosis desalination via real-time membrane surface image analysis. *Desalination* 2010;273:64–71.

6. 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* 2010;20:1261–1269.

7. A. R. Bartman, A. Zhu, P. D. Christofides, Y. Cohen. Minimizing energy consumption in reverse osmosis membrane desalination using optimization-based control. In *Proceedings of the American Control Conference*, 3629-3635, Baltimore, MD, 2010.

8. A. R. Bartman, A. Zhu, P. D. Christofides, Y. Cohen. Nonlinear model-based control and optimization in an experimental reverse osmosis desalination process. In *Proceedings of the American Membrane Technology Association*, accepted, San Diego, CA, 2010.

9. A. R. Bartman, E. Lyster, R. Rallo, P. D. Christofides, Y. Cohen, "Real-Time Image Analysis for Automated Scale Detection and Cleaning in a Reverse Osmosis Desalination System," AIChE Annual Meeting, Nov. 11, 2010, Salt Lake City, UT.

10. A. R. Bartman, A. Zhu, P. D. Christofides, Y. Cohen, "Minimizing Energy Consumption in Reverse Osmosis Membrane Desalination Using Optimization-Based Control," AIChE Annual Meeting, Nov. 10, 2010, Salt Lake City, UT.

11. A. R. Bartman, P. D. Christofides, Y. Cohen, "Optimal Operation and Control of Reverse Osmosis (RO) Desalination," Keynote Session Presentation, Cana-

dian Society for Chemical Engineering (CSChE) Annual Meeting, Oct. 26th, 2010, Saskatoon, SK.

12. A. R. Bartman, A. Zhu, P. D. Christofides, Y. Cohen, "Nonlinear Model-Based Control and Optimization in an Experimental Reverse Osmosis Desalination Process," AMTA Annual Meeting, Jul. 15, 2010, San Diego, CA.

13. A. R. Bartman, P. D. Christofides, Y. Cohen. Nonlinear model-based control of an experimental reverse osmosis water desalination system. *Industrial and Engineering Chemistry Research* 2009;48:6126–6136.

14. 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* 2009;19:433–442.

15. M. Uchymiak, A. Bartman, N. Daltrophe, M. Weissman, J. Gilron, P. Christofides, W. Kaiser, Y. Cohen. Brackish water RO operation in feed flow reversal mode using an EX-situ Scale Observation Detector. *Journal of Membrane Science*, 2009;341:60–66.

16. A. R. Bartman, C. W. McFall, P. D. Christofides, Y. Cohen. Model predictive control of feed flow reversal in a reverse osmosis desalination process. In *Proceedings of the American Control Conference*, 4860-4867, St. Louis, MO, 2009.

17. A. R. Bartman, P. D. Christofides, Y. Cohen. Nonlinear model-based control of an experimental reverse osmosis water desalination system. In *Proceedings of the International Symposium on Advanced Control of Chemical Processes*, paper 145, 6 pages, Istanbul, Turkey, 2009.

18. A. R. Bartman, M. Uchymiak, J. Thompson, H. Gu, A. Zhu, B. McCool, A. Rahardianto, P. D. Christofides, Y. Cohen, "A Novel Approach and System for Rapid Field Evaluation of Water Desalination," AIChE Annual Meeting, Nov. 11, 2009, Nashville, TN.

19. A. R. Bartman, A. Zhu, P. D. Christofides, Y. Cohen, "Modeling and Control of An Experimental Reverse Osmosis Water Desalination System," AIChE Annual Meeting, Nov. 11, 2009, Nashville, TN.

20. H. Gu, A. R. Bartman, P. D. Christofides, Y. Cohen, "Reverse Osmosis Desalination Research in the Undergraduate Laboratory," AIChE Annual Meeting, Nov. 10, 2009, Nashville, TN.

21. A. R. Bartman, P. D. Christofides, Y. Cohen, "Nonlinear Control of a Reverse Osmosis Desalination Process," Southern California Nonlinear Control Workshop, Nov. 6, 2009, San Diego, CA.

22. A. R. Bartman, A. Zhu, P. D. Christofides, Y. Cohen, "Modeling and Control of an Experimental Reverse Osmosis Water Desalination System," North American Membrane Society Annual Meeting, Jun. 20, 2009, Charleston, SC.

23. A. R. Bartman, P. D. Christofides, Y. Cohen, "Nonlinear Model-Based Control of an Experimental Reverse Osmosis Water Desalination System," American Water Works Association, CA-NV Section, Apr. 24, 2009, Santa Clara, CA.

24. C. W. McFall, A. R. Bartman, P. D. Christofides, Y. Cohen. Control of a Reverse Osmosis Desalination Process at High Recovery. *Industrial and Engineering Chemistry Research* 2008;47:6698–6710.

25. C. W. McFall, A. R. Bartman, P. D. Christofides, Y. Cohen. Control of a reverse osmosis desalination process at high recovery. In *Proceedings of the American Control Conference*, 2241-2247, Seattle, WA, 2008.

26. A. R. Bartman, C. McFall, M. Uchymiak, H. Gu, P. D. Christofides, Y. Cohen, "Model Predictive Control of Feed Flow Reversal In a Reverse Osmosis Desalination Process," AIChE Annual Meeting, Nov. 20, 2008, Philadelphia, PA.

27. A. R. Bartman, C. McFall, M. Uchymiak, H. Gu, P. D. Christofides, Y. Cohen, "Model Predictive Control of Feed Flow Reversal in a Reverse Osmosis (RO) Membrane Desalination Process," American Water Works Association, CA-NV Section, Apr. 16, 2008, Hollywood, CA.

28. A. R. Bartman, C. McFall, M. Uchymiak, H. Gu, P. D. Christofides, Y. Cohen, "Model Predictive Control of Feed-flow Reversal in a Reverse Osmosis Desalination Process," American Water Works Association, CA/NV Section Meeting,

Apr. 24, 2008, Hollywood, CA.

ABSTRACT OF THE DISSERTATION

Control and Monitoring of Reverse Osmosis Water Desalination

by

Alex Robert Bartman

Doctor of Philosophy in Chemical Engineering

University of California, Los Angeles, 2011

Professor Panagiotis D. Christofides, Co-Chair

Professor Yoram Cohen, Co-Chair

Water shortages in many areas of the world have increased the need for smarter and more efficient methods for production of drinking water, production of water for agricultural uses, as well as wastewater reuse. In this regard, reverse osmosis (RO) membrane desalination of both seawater and inland brackish water is currently deployed in various locations around the world, with a growing number of large-scale desalination plants in the planning and/or construction stages. In addition, desalination is being increasingly implemented in water reuse applications. RO desalting of agricultural drainage water is also being evaluated for reclamation and reuse of irrigation water.

The design of a water desalination plant is typically tailored to the specific water

source to be desalted necessitating significant field pilot testing, which is a costly and time consuming process. Accomplishing the required pre-plant design tasks (verification of feed pretreatment, anti-scalant dosing, limiting recovery, etc.) in a timely and cost-effective and efficient way is essential. Additionally, in order to meet water production targets while making the process economically and technically feasible, system operation (feed flow rate, feed pressure, overall system recovery) must be maintained at specific operating points which may vary over time due to changing feed water quality. At present, standard operation and control strategies for reverse osmosis membrane water desalination systems do not adequately account for feed water variability and are often operated with suboptimal energy use strategies. Current control strategies (such as directly increasing the feed flow rate to account for decreasing permeate production) which lack intimate knowledge of the process (e.g., the effects of scaling/fouling on the system operation, or the system energy usage) can actually accelerate membrane fouling and cause irreparable membrane/system damage. Optimal operation of RO desalination requires effective process control and energy optimization along with membrane monitoring, which is crucial for inland brackish water desalination, to monitor the state of the reverse osmosis membranes with respect to fouling and mineral salt scaling.

In order to address the above challenges, multiple water purification systems incorporating novel methods for fast evaluation of feed water sources, advanced process control in the presence of high feed water quality variability, system energy

consumption minimization, and automated membrane surface monitoring have been developed, constructed, and tested. These water filtration/desalination systems can produce between 6,000-40,000 gallons of product water per day, and are designed to be highly mobile (small footprint) and modular (for quick reconfiguration). Field tests of these "smart" water systems have been conducted with agricultural drainage water in the San Joaquin Valley, and current field tests include filtration/desalination of cooling tower blow-down water as well as shipboard seawater desalination.

Detailed models of the experimental water treatment systems are developed and used in nonlinear model-based controller design, energy-optimal control, and model predictive controllers. The implemented control algorithms are able to use extensive sensor measurements from the experimental systems (such as flow rates, pressures, conductivities, etc.) to dictate the operating points of the system (e.g., system feed flow rate, feed pressure, overall system recovery). First, model-based control algorithms are derived and implemented to account for transitions between operating steady-states while also mitigating effects on the RO system from disturbances in feed water quality. Next, model-based optimization is conducted to minimize the steady-state energy usage of the RO system. In this way, these control algorithms are shown to improve system operation with respect to set-point transitions, energy usage, feed water flow direction, and also with respect to changes in feed water salinity in order to allow for safe, reliable RO system operation. Additionally, a method for monitoring the progression of mineral salt scaling on RO membrane surfaces is

developed and shown to be effective in mineral scale mitigation (early mineral scale detection and the ability to initiate automated membrane cleaning). The use of this monitoring method to initiate automated membrane cleaning through a process called feed flow reversal is demonstrated in multiple pilot studies and shown to be effective in mitigating mineral salt scaling and recovering RO membrane permeate flux. These concepts are applied to a first generation RO desalination system, and the information gathered is used in the design and construction of a larger, commercial scale, second generation integrated ultrafiltration (UF)/RO desalination system.

Major improvements to system pre-filtration capability (e.g., the addition of a separate, modular pre-filtration process utilizing micro/ultra-filtration with extensive monitoring capabilities), system capacity, and sensor/actuator networks are utilized on the second generation system in order to allow for integrated system operation without the use of an intermediate tank between UF/RO. The improved sensor and actuator networks also facilitate the use of additional robust control strategies for adaptive backwashing/cleaning of the pre-filtration modules, as well as the incorporation of (and the ability to expand on) the control concepts demonstrated on the first generation system. Additionally, the design and implementation of the second generation system improves upon the first generation system by expanding the range of source waters that can be fed into the system, while increasing the overall system capacity and maintaining modular operation with a small system footprint. The design, construction, and usage of the second-generation system is detailed, and a

discussion of future work on the modular research platform is provided.

Chapter 1

Introduction

1.1 Reverse Osmosis Desalination

Water shortages in many areas of the world have increased the need for smarter and more efficient methods for production of drinking water, production of water for agricultural uses, as well as wastewater reuse. Reverse osmosis (RO) membrane desalination has emerged as one of the leading methods for water desalination due to the low cost and energy efficiency of the process [73]. Lack of fresh water sources has necessitated further development of these desalination plants, especially in areas with dry climates. Reverse osmosis is one of the primary means of desalination along with multi-stage flash (MSF), multiple effect distillation (MED), thermal desalting, and others. Out of these technologies, reverse osmosis has been proven to be, in most cases, more environment-friendly, energy efficient, and cost effective [76].

In the process of osmosis (shown in Fig. 1.1(a)), a solvent passes through a semi-permeable membrane in order to equilibrate the solute concentration on both sides. In the case of osmosis in saltwater, the water passes through the semi-permeable

membrane from the region of low salt concentration to the region of high salt concentration. This, in turn, would cause the salt concentrations on either side of the membrane to equilibrate over time. The pressure that forces the water to flow through the membrane is called the osmotic pressure. This osmotic pressure arises from the differences in fugacity of the solvent on either side of the semi-permeable membrane as represented below (assuming both sides of the membrane are at equal pressures, $P^1 = P^2$):

$$\hat{f}_2(T, P^1 = P^2, x^1_{solv} < 1) < \hat{f}_2(T, P^2, x^2_{solv} = 1) \tag{1.1}$$

where $\hat{f}_2$ is the fugacity of the solvent, T is the temperature (in K), P^1 and P^2 (in Pa) are the pressures in compartments 1 and 2 on either side of the membrane, respectively, and x^1_{solv} and x^2_{solv} are the mole fractions of the solvent in compartments 1 and 2. This difference in fugacity represents a driving force for mass transfer, and the solvent diffuses through the partition [80]. In order to establish equilibrium, P^1 must be adjusted to an appropriate value, $P^\star$. The osmotic pressure difference ($\Delta\pi$), which is a measure of the chemical potential difference between the solution on the feed and permeate side of the membrane, can be expressed fundamentally in terms of this required pressure increase as:

$$\Delta\pi = P^\star - P^1 \tag{1.2}$$

Applying a Poynting factor (to account for the pressure effect on fugacity of a
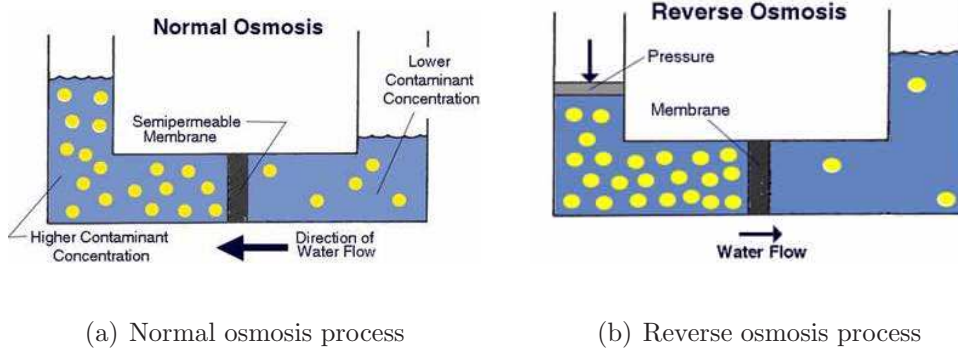
(a) Normal osmosis process   (b) Reverse osmosis process

Figure 1.1: Normal and reverse osmosis processes.

species in solution), and in the limiting case that the solution is sufficiently dilute with the respect to the solute (see [80] for the detailed derivation), the osmotic pressure can be approximated through the following relation:

$$\Delta \pi = \phi \Delta n R T \tag{1.3}$$

where ϕ is the dimensionless van't Hoff factor, $\Delta \pi$ is the osmotic pressure across the membrane (in Pa), Δn is the difference in salt concentrations on either side of the membrane (in mol/m^3), R is the ideal gas constant (in $m^3 Pa \ K^{-1} mol^{-1}$), and T is the temperature (in K) [34]. For more concentrated solutions, relations such as the one given in Eq. 3.17 are utilized for estimation of the osmotic pressure difference.

In reverse osmosis (RO) (shown in Fig. 1.1(b)), a pressure is applied in the direction opposing the osmotic pressure. If this applied pressure is equal to the osmotic pressure, no solvent will flow through the membrane. If this pressure becomes higher than the osmotic pressure, the reverse effect is observed; solvent will flow

from the region of high salt concentration to the region of low salt concentration. This process will cause the salt on the high concentration side to become even more concentrated, whereas the salt concentration on the other side will decrease.

The solution-diffusion model is commonly used to describe the transport of solute/solvent through the RO membranes [90]. In this model, the solvent (water) transport through the membrane is represented by:

$$J_v = L_p(\Delta P - \sigma \Delta \pi) \tag{1.4}$$

where J_v is the water permeate flux (in m^3 solvent/m^2 membrane area/s), L_p is the hydraulic permeability (in $m/Pa/s$), ΔP is the pressure difference (in Pa) between the feed and permeate side of the membrane, and σ is the reflection coefficient which represents the selectivity of water passage relative to salt passage.

The solute (salt) flux equation is a combination of a diffusive flux through the membrane (which depends on the concentration driving force) and a term representing convection (which depends on the permeate flux). The equation for solute flux is given below:

$$J_s = B \Delta c + (1 - \sigma')c_m J_v \tag{1.5}$$

where J_s is the solute flux (in kg/s), B is the solute permeability (in m/s), Δc is the concentration driving force (in mol/m^3), σ' is the solute reflection coefficient, and c_m
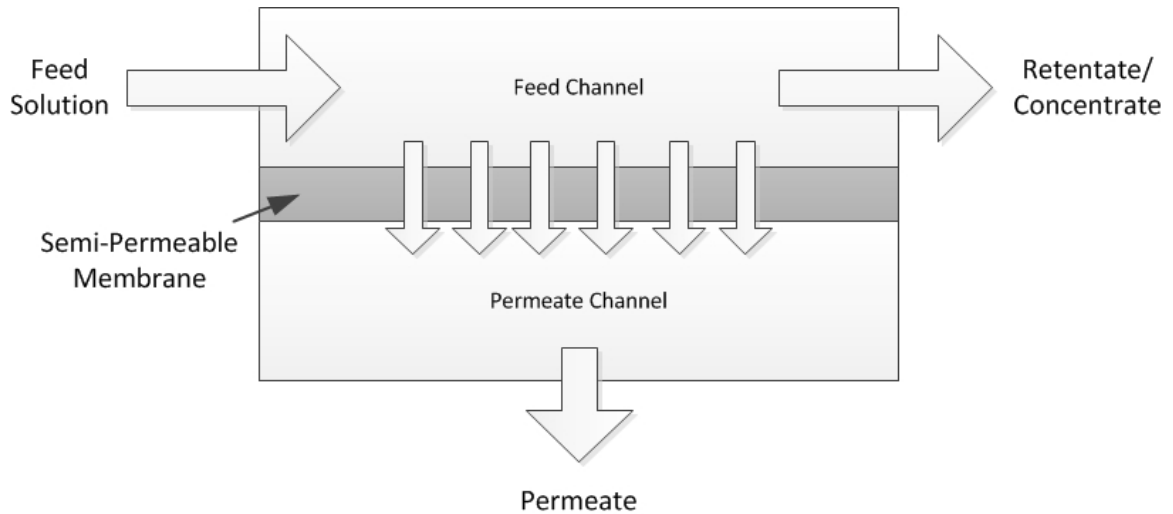
Figure 1.2: Schematic diagram of a cross-flow filtration process.

is the solute concentration at the membrane surface (in mol/m^3). However, for high-rejection RO membranes, this solute reflection coefficient (σ') is approximately equal to unity, leading to the cancelation of the convective term for typical RO applications [82].

RO membrane systems utilizing flat-sheet membranes (membranes can also be found in hollow-fiber or tubular form) (as shown in Fig. 1.2) are commonly found as a plate and frame unit, or (in most commercial applications) as a spiral wound unit [60]. In high-throughput RO processes, a cross-flow configuration is used. In this configuration, water is pressurized by a pump and fed into the RO pressure vessel (either a plate and frame unit, or a pressure vessel containing a spiral-wound membrane). As the feed water flows through the feed channel, permeation of water through the RO membrane occurs. Permeate water is collected, while the remaining

high-concentration stream ("concentrate", or "brine" stream) is removed from the tail end of the vessel for disposal.

In cross-flow membrane systems with flat sheet RO membranes, issues with precipitation of solutes may occur. As the feed solution flows through the feed channel, permeation of the solvent through the semi-permeable membrane carries the solute ions toward the membrane surface. Since the purpose of the membrane is to reject these solute ions (such as Na^+ and Cl^- in seawater RO membrane desalination), an accumulation of these ions occurs at the membrane surface. This phenomenon is called "concentration polarization" (or "CP"), and is represented in Fig. 1.3. The concentration polarization modulus (CP) is often estimated by the simple film model equation:

$$CP = \frac{c_m}{c_b} = (1 - R_0) + R_0 \, exp\left(\frac{J_v}{K_m}\right) \tag{1.6}$$

where c_m is the concentration at the membrane surface (in mol/m^3), c_b is the bulk concentration of the solute (in mol/m^3), K_m is the feed-side solute mass transfer coefficient (in m/s), and R_0 is the dimensionless observed rejection given as $R_0 = 1 - c_p/c_f$, where c_p is the permeate solute concentration (in mol/m^3) and c_f is the feed solute concentration (in mol/m^3).

It can also be seen that concentration polarization increases as the feed solution travels down the length of the feed channel, due to the fact that the concentration of the solute in the bulk is increasing at a slower rate than the accumulation of

the solute ions at the membrane surface. This bulk increase in solute concentration occurs because the solvent is being removed as permeate (through the semi-permeable membrane). As a consequence of concentration polarization, the solvent flux (or permeate flux) is decreased down the length of the feed channel due to the rise in osmotic pressure at the membrane surface. While the decreased permeate flux is problematic in RO membrane desalination operations, larger issues arise when the solute reaches concentration levels where precipitation can occur on the membrane surface (mineral scaling) or in the bulk solution. These issues require careful control of RO system operation, such as the adjustment of the feed flow rate and overall system recovery to maintain concentrations of the solute ions below the precipitation thresholds, as well as advanced monitoring processes to detect the onset of mineral scaling. Mineral salt scaling and its effect on RO system operation is discussed in greater detail in Chapter 6.

In the work presented in this thesis, the experimental systems utilize spiral-wound membrane units to conduct high-throughput RO membrane water desalination. Spiral-wound membrane units (as shown in Fig. 1.4) operate similarly to plate and frame units, except in these units, the membrane is rolled into a tight spiral (sandwiched between feed/permeate spacers in order to preserve a feed/permeate channel) in order to increase the amount of active membrane surface area per volume. The feed stream is introduced through one end, the brine stream outlet is on the other end, along with the permeate stream outlet (the permeate is collected in the middle
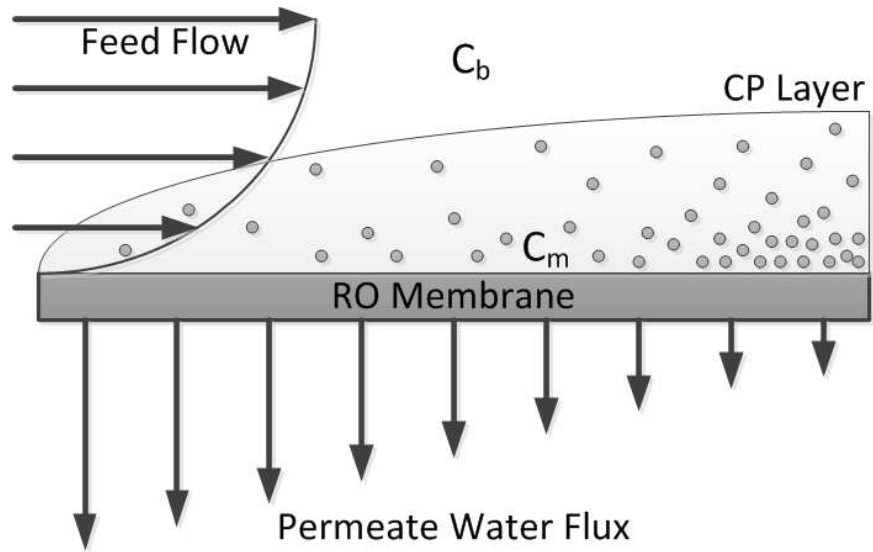
Figure 1.3: Schematic representation of concentration polarization. The feed solution flows through the feed channel from left to right over the semi-permeable membrane, while the solute particles accumulate at the membrane surface due to solute rejection by the membrane.
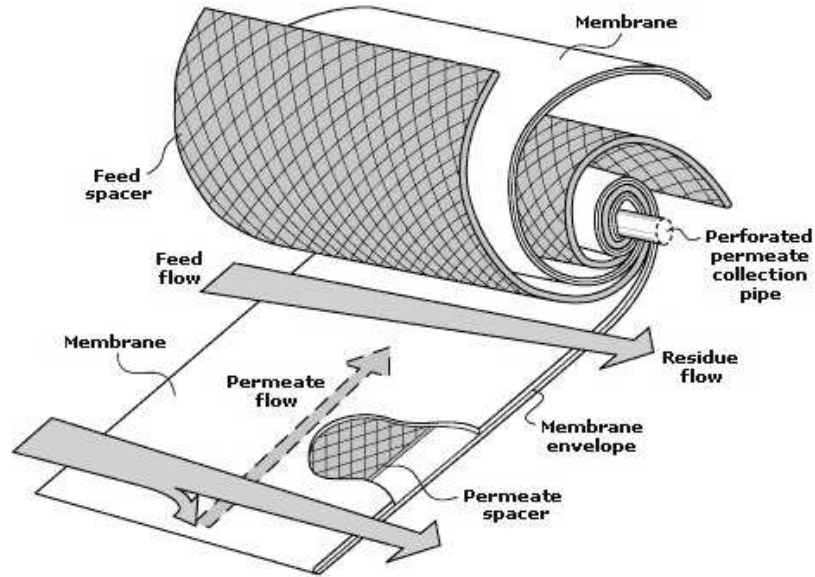
Figure 1.4: Schematic diagram of a spiral-wound membrane module (courtesy of Membrane Technology Research (MTR)).

of the spiral). In industrial systems, these spiral wound units are typically 4 or 8 inches in diameter, and 40 inches long.

In most RO desalination systems, the water is pre-treated (i.e., with cartridge filters, micro/ultra/nano-filtration, etc.) to remove any large particles, bacteria, or other biological materials. The treated water is then pumped to high pressure and fed to the membrane unit(s). The permeate stream is commonly post-treated to remove any additional impurities or after-effects caused by the pre-treatment, and the brine stream is managed by disposal, further volume minimization, or reuse in suitable high-salinity applications.

1.2 Background on Control of Reverse Osmosis Water Desalination

Even with advances in reverse osmosis membrane technology (membranes with higher permeability, fouling/scaling resistance, etc.), maintaining the desired process conditions is essential to successfully operating a reverse osmosis desalination system. Seasonal, monthly, or even daily changes in feed water quality can drastically alter the conditions in the reverse osmosis membrane modules, leading to decreased water production, sub-optimal system performance, or even permanent membrane damage [5, 25].

In order to account for the variability of feed water quality, a robust process control (for the key system operating parameters such as feed flow rate, feed pressure, overall system recovery, etc.) strategy that takes into account desired system operating conditions, feed water quality, system operational limitations, and in some cases, feed water scaling propensity/energy usage is necessary. In a modern reverse osmosis (RO) plant, automation and reliability are elements crucial to personnel safety, product water quality, meeting environmental constraints, and satisfying economic demands. Industrial reverse osmosis water desalination processes primarily use classical proportional (P) and proportional-integral (PI) control to monitor production (permeate) flow and adjust feed pumps or permeate back-pressure valves accordingly

[9]. While such control strategies are able to maintain a consistent product water (permeate) flow rate, they may fail to provide an optimal closed-loop response with respect to set-point transitions owing to the presence of nonlinear process behavior [5, 25]. Additionally, these traditional control strategies fail to comprehensively account for temporal changes in feed water quality. In some cases, permeate production can decrease due to scaling or fouling on the membrane surface. When this occurs, traditional control algorithms force the system to increase overall system recovery (and in turn, permeate flux) which can lead to an increased rate of scaling, irreversible membrane damage, and eventual plant shutdown. Traditional process control schemes are also unable to monitor plant energy usage and make adjustments toward energy-optimal operation.

Early automatic control methods utilize model based control based on a linear model [10]; using step tests to create a model that is a linear approximation around the desired operating point. In [10], the authors use a data-based model formulation to develop closed-loop control systems to control permeate flux and permeate conductivity. The manipulated variables (inputs) are the feed pressure and feed pH, which are each linked in a single-input single-output (SISO) fashion to the outputs (feed pressure to permeate flux, and feed pH to permeate conductivity). This method is useful for a specific RO system where the model data has been gathered, but would not be useful for application to any other types of RO system. Also, the model does not take into account any coupled effects, such as the effect of feed pressure on the

permeate conductivity.

In addition to classical control schemes and linear formulations, nonlinear model-based geometric control strategies have been developed to minimize the effects of varying feed water quality and also to account and correct for various faults that may present themselves during the operation of a reverse osmosis desalination process [16, 63] (see Chapter 3 and Appendix A for more information on this work). Model-based control is a promising alternative to traditional RO plant control strategies. Several model based methods such as model-predictive control (MPC) and Lyapunov-based control have been evaluated via computer simulations for use in reverse osmosis desalination [3, 25, 33, 44, 6]. The authors of [3] demonstrated that the use of a dynamic matrix control (DMC) algorithm in a MPC framework shows improvement in the control of permeate flow rate and conductivity in large set-point transitions when compared to classical PI control. A data-based, linear model is used to predict the permeate flow rate and permeate conductivity through changing the feed pressure and feed pH. However, this model does not utilize any analytical models of RO operation, and is only applicable to a specific system where the initial data to derive the linear model was gathered. Using this process would require extensive data collection and modeling before this approach could be used on a different system. In [44], the authors utilize a MPC framework (based on work presented in Appendix A to operate the plant at optimum operating conditions (flow rates/pressures) to decrease water production cost. The work of Appendix A and Chapter 5 is extended to several

case studies on seawater RO (SWRO) plant operation to analyze the economics of the process. The authors in [6] used an analytical model of RO operation in a MPC framework where feed pressure and concentrate flow rate are manipulated to control permeate flow rate and concentration. Again, the authors use work presented in this thesis (Appendix A and Chapter 4) to guide the implementation of the MPC algorithms and RO process models. This work also builds on the work in [3] by changing one of the manipulated inputs from feed pH to brine stream flow rate in order to relate better to industry standards for RO control. These works use similar mathematical MPC implementation, but only use a small number of variables for manipulated inputs (feed pressure, concentrate flow rate, feed pH), and are not as extensive as the control formulations presented in this thesis. Also, the controllers presented in these works were not directly tested on experimental systems, unlike the experimentally-verified controllers presented in this thesis.

Experimental system identification and MPC applications have also been previously attempted [14, 21, 8]. These works use similar MPC formulations to the works above, but are able to compare to experimental data to verify their control performance. In [14], the authors use an MPC-based controller similar to [3] (using a data-based model with inputs of concentrate flow rate and inlet acid dosing, and outputs of permeate flow rate, permeate conductivity, trans-membrane pressure, and inlet pH). Through experimental testing, the authors were able to verify the benefits of using the constrained MPC (CMPC) algorithms over classical PI control for con-

ducting set-point changes in the output variables. Again, the data-based model will only perform well on one very specific RO system, and would need to be reconfigured for each new RO system due to its empirical nature. The authors of [21] also use a CMPC formulation, but conduct their experiments on a hollow-fiber membrane module system. The authors show that the CMPC performance is better than the classical PI control in similar fashion to the work in [3]. The work in [8] uses linear regression and multiple-input single-output (MISO) formulation to develop empirical models for recovery ratio, permeate quality (TDS), and power consumption using inputs of feed flow rate, feed quality, feed pressure, feed pH, and feed temperature. The authors demonstrate that the predicted outputs match closely with simulation values from the ROSA software, and further conduct a sensitivity analysis to see which inputs have the largest effect on the model outputs. These works attempt to verify the model/controller performance based on comparison with experimental system data or commercially available simulation software (ROSA), but still utilize empirical models for their mathematical system formulations.

Various first-principles models for aspects of RO systems have also been proposed, such as [45] for predicting concentration polarization and permeate concentration, but there are fewer models for overall RO system operation (such as [63], presented in Chapter 3). The authors of [45] begin with many of the same solution-diffusion model equations as presented in Eqs. 1.4 - 1.5, but focus on the modeling of concentration polarization. Through mass balances around a feed/product tank and the membrane

modules, the authors developed a non-linear model to predict concentration in the feed channel, as well as permeate concentration. The authors then test this model using experimental data from the literature to show that the model can accurately predict the effects of concentration polarization. This model is useful for predicting the concentration polarization, but does not predict the actuator settings required for specific system operating points (feed flow rates, system pressures, overall system recovery) which are needed for overall RO system control.

Model based control methods have also been used in conjunction with fault detection and isolation schemes to improve robustness of control methods in the presence of sensor and actuator failures [24]. In this work, the authors used the model presented in Chapter 3 to provide a reference value for the retentate stream flow rate in order to detect actuator faults in the retentate stream valve. It was shown that if the model can accurately represent the system, actuator faults can be quickly detected and can even be mitigated if the system has a fall-back configuration (in this case, a redundant valve in parallel with the actuator in question is used to resume normal system operation in the event of a fault).

Extensive research has also been completed in using fuzzy logic, neural networks, genetic algorithms and probabilistic reasoning for the purposes of RO system control [101, 4, 52]. In [101], the authors evaluate the effectiveness of a neural network and fuzzy logic control system on a SWRO pilot plant. In this work, the controller is able to take in a greater number of inputs (temperature, feed TDS, feed pH, feed flow

rate, feed pressure, concentrate pressure, permeate flow rate, permeate conductivity, salt rejection, recovery, scale index) and use these in a data-based neural network to make actuator control decisions (such as opening/closing the retentate valve). The authors show that the system is able to achieve a constant recovery of 30% and maintain high rejection, while reducing the manpower requirements and overall chemical consumption of the plant. The authors of [4] use a neural network model based on feed pressure, feed temperature, and feed salt concentration to predict the permeate flow rate. The authors showed that the neural network model was able to predict the permeate flow rate with correlation coefficients of 0.989 to 0.998 (the higher value corresponds to when the model input data was taken from data actually used to train the model, while the lower number represents the case where the model input data was not used in the neural-network model training) when compared with the actual experimental data. However, the authors state that while the method is accurate for interpolation, its use is very limited when the model inputs fall outside of the range of the training data (model extrapolation). The work in [52] is very similar; using neural networks with inputs of feed temperature, feed TDS, trans-membrane pressure, feed flow rate to predict permeate flow rate and permeate quality. Data from experimental plant operation was used to train the model, and the model testing yielded fairly accurate results (correlation coefficients of 0.96 for permeate TDS and 0.75 for permeate flow rate). These more advanced data-based methods show a more comprehensive approach to overall RO system control than the data-based linear/non-

linear MPC models, but have many limitations when model extrapolation is necessary (due to the nature of training of the neural networks).

Furthermore, with the rising cost of energy, it is also desired to find operating methods to reduce the energy consumption of reverse osmosis desalination processes in the presence of feed water variability [35, 69]. In [35], the authors cite the work from Chapter 4 in order to utilize a RO model for the purposes of RO system energy minimization. Various input parameters are taken into account, such as membrane permeability, concentration polarization, temperature effects, membrane fouling, and VFD efficiency, in order to use mixed-integer non-linear programming (MINLP) to predict plant energy usage. The authors use a mostly analytical model with several correlations, and also examine the effects of time-varying electricity cost on the overall system operation cost. It was determined that substantial cost savings can be achieved by operating the system during specific off-peak hours. More information regarding energy-optimal system operation can be found in Chapter 5.

Other control methods have also been evaluated in the context of RO system integration with renewable energy sources [43, 54, 20, 42, 55, 20, 36]. These works focus mostly on the integration of the RO processes with renewable energy infrastructure (PV panels, inverters, batteries), and do not focus on the operational control of the RO process with respect to energy usage/scaling/fouling. With respect to results in the broad area of the mathematics of optimization-based control, the reader may refer to the following papers for results on real-time optimization [93, 18], self-optimizing

control [79], and extremum-seeking control [49].

1.3   Dissertation Objectives and Structure

The objective of this dissertation is to develop advanced ("smart") methodologies for reverse osmosis (RO) desalination system control, optimization, operation and monitoring, and to evaluate their performance on experimental commercial-scale RO processes. The controllers presented utilize an extensive sensor network (multiple flow rates, pressures, conductivities, pH measurements, turbidity measurements) in order to operate the pre-filtration/RO system at optimal conditions (feed flow rate, feed pressure, overall system recovery) while accounting for changes in feed water quality, system energy utilization, and membrane fouling/scaling. This dissertation provides a theoretical framework that allows for the development of a comprehensive RO system operational model and the derivation of various model-based controllers for safe, reliable, and energy-optimal system operation.

The development of advanced control algorithms for RO desalination systems is critical to addressing the need for the future development of fresh water supplies, as well as to upgrade existing systems to allow for higher yield and more cost-effective operation. The results presented in this work indicate the viability of several types of RO system control and optimization, both for overall system control and for specific applications such as membrane cleaning/mineral salt scaling prevention.

The dissertation also provides a background on the concepts, design, construction, and testing of the experimental water purification systems built at UCLA. The advantages and novel concepts of the first generation RO membrane desalination are presented, along with a detailed description of the system components. The second generation UF/RO water purification system is also detailed, with discussion of the reasoning behind its construction as well as the process/control improvements over the first generation system.

The structure of this thesis is as follows: first, the first-generation experimental RO water desalination system developed at UCLA (M3 system) is introduced in Chapter 2. This system provides the platform for implementing the control and optimization methodologies presented in the later chapters (Chapters 4 - 6). The overall concept behind this experimental system and its advantages are discussed, and the system components are detailed. Specific details are then provided on the electrical and control systems.

Chapter 3 explores the first-principles mathematical modeling and model verification that was carried out in order to allow for the derivation of the advanced control algorithms implemented on the first-generation M3 system. Modeling is conducted using mass and energy balances around a simplified version of the RO system, taking into account the action of the actuated retentate valve, the feed flow rate (through the action of the variable frequency drive), and also the actuated bypass valve. This derivation is followed by a description of the model verification carried out on the

M3 system in order to fit the various model parameters to the actual experimental system operation data.

Once the mathematical model of the experimental M3 RO system was derived and verified, process control algorithms were designed to address common problems associated with RO system operation such as the effects of feed water quality disturbances and the optimization of the RO system energy usage. In Chapter 4, nonlinear model-based feedback control of the experimental M3 RO system is presented and the performance of this controller is evaluated for use in situations where large set-point changes in retentate stream flow rate are induced. The controller uses the system model (in real-time) to calculate the appropriate feed flow rate set-point and actuated retentate valve position for maintaining system operation at the desired pressure and recovery. The controller is also tested for its ability to reject very large disturbances in the feed salinity, which are a common problem in commercial/industrial RO plant operation. Practical implementation of the controller on the experimental RO system is also discussed.

Chapter 5 describes a model-based optimization algorithm for energy minimization of RO processes. The derived system model is used in conjunction with a theoretical formulation for specific energy consumption (SEC) as a metric to evaluate the energy usage of the experimental RO system. The optimization-based controller is implemented on the M3 system and experiments are carried out to determine the performance of the controller, as well as the deviation between the predicted energy

usage and the experimentally determined energy usage of the M3 system. The controller again uses the system model to predict and implement the optimal feed flow rate and retentate valve position in order to minimize the system energy consumption in situations where the feed water quality changes.

In Chapter 6, novel image based control of a RO membrane scale mitigation process called feed flow reversal (FFR) is developed and implemented for the first time on the experimental M3 system. Mineral salt scaling is a common issue in RO processes (mostly in inland brackish water desalination) and can occur when the RO systems are operated at high recovery with feed water of high scaling propensity. This mineral scale can decrease the permeate flux in the RO membrane modules and can lead to irreversible system/membrane damage. Using the novel image analysis software, a method for determining the appropriate amount of time before initiating RO membrane cleaning (through feed flow reversal) using an ex-situ membrane monitor (MeMo) coupled with automated image analysis software is evaluated. The experimental set-up is described, and the novel image analysis algorithm is detailed. Results from several experiments (some utilizing anti-scalant chemicals) are presented to confirm the effectiveness of the automated detection in tracking the fractional surface coverage of mineral salt scaling, as well as the number of scaling crystals present in the captured image. Multi-cycle FFR operation is also detailed, demonstrating the effectiveness of the FFR process when used over long operation periods in an automated fashion.

The laboratory and field experience with the M3 system suggested areas of system and operational control improvements which were implemented in a second-generation water treatment system. Improvements included (but are not limited to) adaptive pretreatment, expanded sensor/actuator network, and a larger fresh water production capacity. Chapter 7 describes the concepts and design of this second generation (CoM2RO) commercial scale RO desalination system at UCLA. This chapter also details the capabilities of the CoM2RO system and the improvements made over the first-generation M3 system. The major components of the system, and specifically the system electrical/control aspects are discussed in detail. The chapter also discusses the novel aspects of system control and operability made possible through knowledge gained from field deployment of the first-generation (M3) system.

Chapter 8 summarizes the main conclusions of this dissertation.

Several appendices are also included with supplementary reference materials:

Appendix A demonstrates a theoretical method for using model predictive control (MPC) for system component protection when switching to and from a membrane cleaning process called feed flow reversal (FFR). A specialized case of the system model presented in Chapter 3 with a bypass valve (to control feed flow rate, instead of a variable frequency drive on the feed pump), is used to simulate the process conditions (feed pressure, bypass/retentate/permeate flow rates) when the switch to feed flow reversal is initiated. This controller serves to direct the actuated retentate and bypass valves in such a way that prevents water hammer in the system. The

differences between using MPC and classical control methods such as proportional-integral control (PI) are also evaluated.

Appendix B presents a listing of the sensors and actuators used in the first (M3) and second (CoM2RO) generation RO systems, along with a list of control system hardware.

Appendix C presents the MATLAB code used for the calculations in Chapters 4 - 6 and Appendix A .

Chapter 2

First-generation System (M3): Design and Construction

2.1 Overall Concept and Design

The experiments presented in this thesis were carried out using UCLA's first-generation M3 (mini, modular, and mobile) RO desalination system. The concept behind this system was to design and construct a RO water desalination system that could be easily reconfigured for almost any type of source water, and could be loaded into an average cargo van for quick transportation to field locations. The modularity of the M3 design also allows for entire modules (pre-filtration, pumping, membranes, post-filtration, scaling detection) to be added or removed quickly and easily. The M3 system also includes a versatile monitoring and control system including multiple sensors, actuators, and embedded computing for the implementation of advanced control algorithms.

With respect to this thesis, the M3 system provides a test platform for testing

advanced control strategies on a novel embedded computing system with the ability to conduct real-time model-based control. The M3 system is able to process data with more complex algorithms than traditional programmable logic controllers (PLCs) due to the highly-customizable nature of the embedded computing platform. The extensive network of sensors and control provided by the system actuators also allows for the implementation and immediate testing of advanced control algorithms such as model predictive control and optimization-based control.

The M3 experimental system (shown in Fig. 2.1 and in the photograph in Fig. 2.2) is comprised of a feed tank, two low-pressure centrifugal feed pumps in parallel (JM3460-SRM, Sea Recovery, Carson, CA, utilizing 208VAC, 3-phase power) which pass the feed water through a series of cartridge filters while also providing sufficient pressure for operation of the high-pressure pumps (requiring greater than approximately 15 psi at the inlet), two high-pressure axial-piston (positive displacement) pumps in parallel (Danfoss CM-3559, Baldor Reliance Motor, Carson, CA, each capable of delivering 4.2 gallons per minute at 1000 psi, also utilizing 208VAC, 3-phase power), and a bank of 18 pressure vessels (3" diameter, Sea Recovery, Carson, CA) containing spiral-wound RO membranes (Dow Filmtec XLE2540) . The high-pressure pumps are outfitted with variable frequency (or variable speed) drives (one per high-pressure pump, Teco Fluxmaster FM50) which enable the control system to adjust the feed flow rate by using a 0-10V output signal. The bank of 18 membranes are arranged into 3 sets of 6 membranes in series; and for the control experiments pre-

sented in this work, only one bank of 6 membrane units was used. The experimental system uses solenoid valves (GC Valves, HS4GF15A24) controlled by the data acquisition and control hardware to enable switching between multiple arrangements of the membrane modules (2 banks of 6 in parallel to one bank of 6 in series, or any number of the modules in series) while also allowing for control of the flow direction through the membrane banks. After the membrane banks, an actuated valve (ETI Systems, VA8V-7-0-10) is used to control the cross-flow velocity (v_r) in the membrane units, while also influencing system pressure. This valve is used as an actuator for the control system utilizing the control algorithms presented in Chapters 4, 5, and 6. During the experiments presented in these chapters, the resulting permeate and retentate streams are fed back to the tank in an overall recycle mode (for field operation the system can be operated in a one-pass fashion where the retentate stream is discarded to a drain as waste and the permeate stream is collected for use). The M3 system utilizes 304 and 316 alloys of stainless steel in the high-pressure sections (for wetted sensor/actuator components, as well as piping) to prevent corrosion of system materials, and utilizes flexible PVC tubing in the low pressure sections.

The M3 RO system also has an extensive sensor and data acquisition network; flow rates and stream conductivities are measured in real-time for the feed stream, retentate stream and permeate stream. The pressures before each high pressure pump, as well as the pressures before and after the membrane units (feed pressure and retentate pressure) are also measured. The system also includes sensors for measuring

feed pH, permeate pH, in-tank turbidity, and feed turbidity after filtration (in real-time). A detailed listing of components is provided in Table B.1. A centralized data acquisition system takes all of the sensor outputs (0-5V, 0-10V, 4-20mA) and converts them to process variable values on the local (and web-accessible) user interface where the control system is implemented. Using the system data in real-time, the stability, performance, and robustness of various control methodologies can be tested (see Chapters 4, 5, and 6 for additional details regarding experimental set-up and control algorithms). The data is logged on a local computer as well as on a network database where the data can be accessed via the internet, while the control portion of the web-based user interface is only available to persons with proper authorization. The data acquisition and control system uses National Instruments software and hardware to collect the data at a sampling rate of 10 Hz and perform the necessary control calculations needed for the computation of the control action to be implemented by the control actuators (see separately provided thesis addendum for details about the embedded controller code).
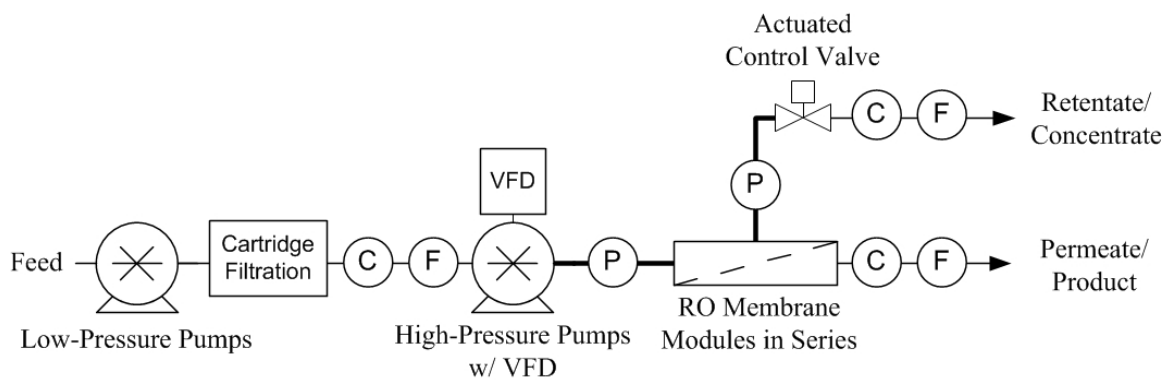
Figure 2.1: M3 RO membrane water desalination system schematic. Circles denote sensors (P = pressure, C = conductivity, F = flow). Cartridge filtration consists of a 25 micron filter and a 5 micron filter in series.
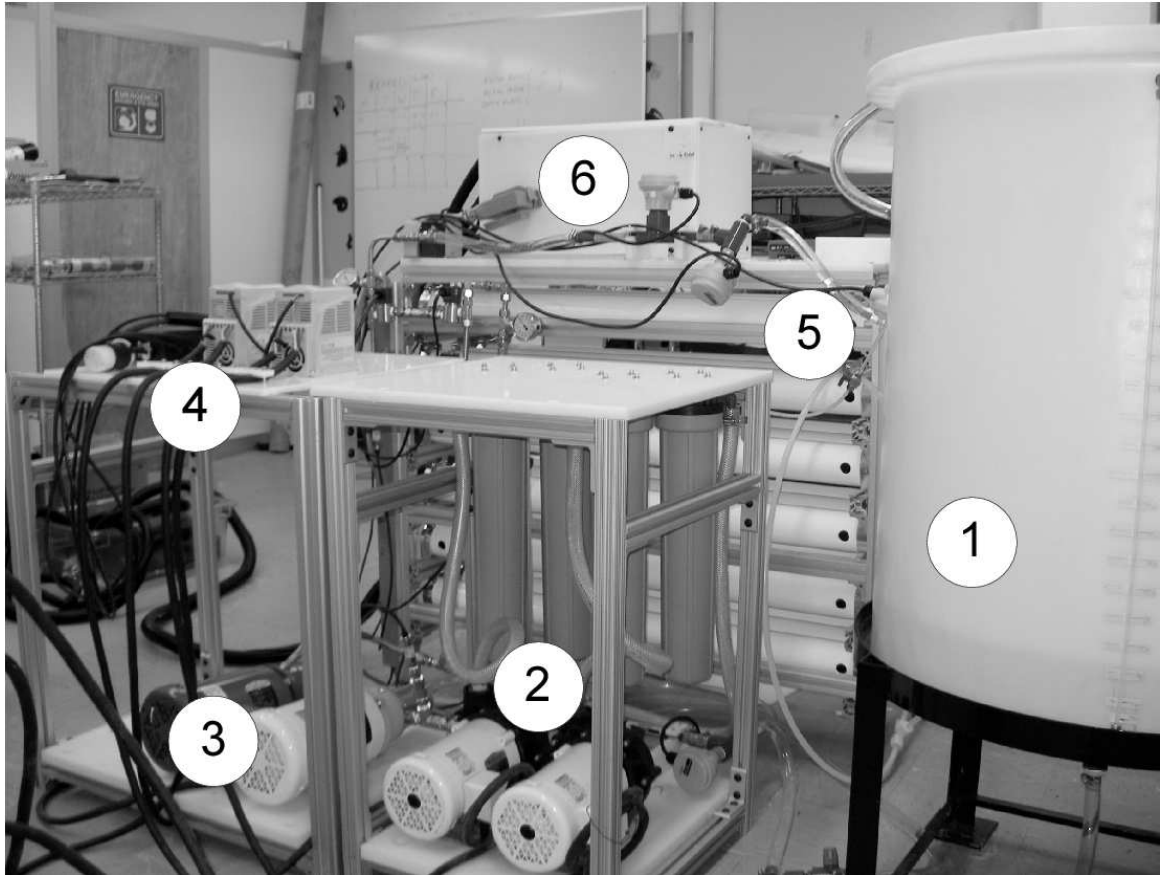
Figure 2.2: UCLA experimental RO membrane water desalination system: (1) Feed tank, (2) Low-pressure pumps and prefiltration, (3) High-pressure positive displacement pumps, (4) Variable frequency drives (VFDs), (5) Pressure vessels containing spiral-wound membrane units (3 sets of 6 membranes in series), and (6) National Instruments data acquisition hardware and various sensors.

2.2 Electrical and Control Systems

The sensors on the M3 RO system output either 4-20mA or 0-5/10VDC. Standard 2/3-wire sensor connections are utilized, and the corresponding wiring diagrams are depicted in Appendix B . 500-ohm resistors are also used to convert the 4-20mA signal to a 0-10VDC signal for measurement by the controller modules. Power for these sensors is provided by 12/24VDC (output) power supplies which use 115VAC from typical wall outlets as input power. These power supplies include internal fuses for safety in the event of a short circuit or failure in the system wiring. Internal to the control enclosure, sensors and actuators are connected through the use of terminal blocks (DIN rail mount) and bus-bars for modularity in power and signal distribution. For external wiring, heavy-duty three-pole connectors are used for the majority of the sensors, along with strain relief on the sensor-end wire connections. Some of the sensors added later in construction (such as the current/voltage sensors for VFD power consumption measurement) utilize smaller two/three pole plugs due to lower current flow and space-saving considerations.

M3 system actuators (variable frequency drives, actuated retentate valve) were selected to utilize a 0-10VDC control signal, or to operate through relay control (on/off actuators such as solenoid valves). In the case of the solenoid valves, a cascade of two relays was utilized due to the low maximum current allowed through the control system relay module channels. Wiring diagrams for the system actuators are also

detailed in Appendix A. Wiring connections are implemented in a similar manner to the sensors, with modular plug/receptacle connectors and strain relief for direct wiring connections.

The M3 control system consists of a stand-alone dedicated real-time controller connected to a field-programmable gate array (FPGA) for high-speed data input/output (I/O) to the various types of I/O modules (component details can be found in Table B.2). The M3 controller I/O modules include input voltage measurement for sensor measurement (-10 to 10VDC), voltage output for actuator control (0 to 10 VDC), and mechanical relay modules (on/off switches) for solenoid valve control. The stand-alone controller is connected to a fanless, rugged PC (Tangent, Mini Medallist Pro), which is used as the user interface and also for advanced control algorithm implementation in various programming languages (MATLAB, LabVIEW, etc.). The graphical user interface (GUI) code is implemented in LabVIEW on the rugged PC, and is deployed on the real-time control target. FPGA code is also used to provide high-speed I/O to the individual modules on the embedded controller chassis, as well as providing basic safety checks (high pressure, etc.). The M3 FPGA/GUI code is provided with detailed description in a thesis addendum, provided separately.

Chapter 3

Mathematical Modeling and Model Verification

3.1 Mathematical Model

In order to utilize model-based control on the M3 RO desalination system, a mathematical model of the system (to determine system pressure and system flow rates at given actuator conditions) was derived. The dynamics of the M3 system are modeled based on the use of mass and energy balances [19]. First, an energy balance was conducted around the actuated valve; then an overall mass balance was derived in order to obtain the dynamic equations representing the changing velocity in the various streams with respect to time.

As seen in the simplified system diagram (Fig. 3.1), feed water enters the system and is pressurized by the high-pressure pump (equipped with a variable frequency drive for feed flow rate control). The pressurized stream is then fed to the spiral-wound membrane unit(s). Two streams exit the membrane module, the retentate
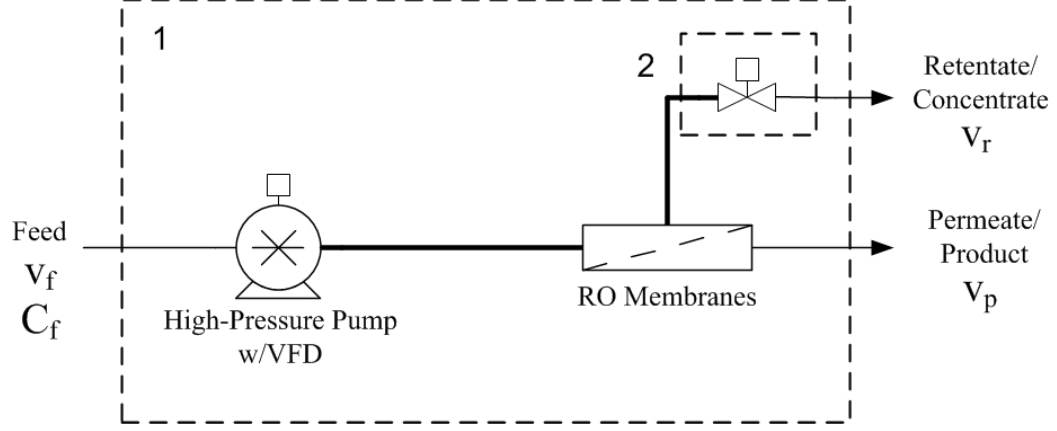
Figure 3.1: Simplified M3 process diagram showing control volume boundaries (1, around the entire system and 2, around the actuated retentate valve) for use in model derivation.

(or brine) stream, with velocity v_r, and the permeate stream (v_p). The downstream pressure of all of the exit streams is assumed to be atmospheric pressure. Initially, a simplified model is derived as presented below:

In order to determine the effect of the valve actuator position on the flow rates and pressures of the system, a macroscopic mechanical energy balance is taken around the retentate stream valve (control volume 2) and can be written as [19]:

$$\frac{d}{dt}(K_{tot} + \phi_{tot}) = -\Delta(\frac{1}{2}\frac{<v^3>}{<v>} + \hat{\phi} + \frac{p}{\rho})w + W_m - E_c - E_v \tag{3.1}$$

where the first term represents the balance of kinetic energy entering and leaving the control volume ($<v>$ is the average fluid velocity over the pipe cross-section (in m/s), $\hat{\phi}$ is the potential energy difference between the inlet and outlet (in J/s), p is

the pressure difference between the inlet and outlet (in Pa), ρ is the density of the fluid (in kg/m^3), and w is the mass flow rate of the fluid (in kg/s)). The second term (W_m) represents the rate of doing work on the fluid by moving surfaces in the control volume (in J/s), the third term (E_c) is the compression term (in J/s), and the final term is the viscous dissipation (or friction loss) term (in J/s).

In order to simplify the problem, several assumptions can be invoked. The first assumption is that the fluid is incompressible, which is reasonable for the liquid system in question. Secondly, it is assumed that all components of the system are in the same horizontal plane, negating any potential energy terms due to gravity. Next, it can be seen that there is no external work done on the fluid inside the energy balance boundary (containing only the pipe and valve). The fourth assumption states that the density of the water is a constant at all of the boundaries in the energy balance, which leads to the fifth assumption, stating that the velocity at the entrance and exit planes of the boundary (control volume 2 in Fig. 3.1) is equal. This is reasonable since there is no accumulation; all the mass entering the pipe before the valve exits the pipe after the valve. If the mass flow is equal at both boundaries, and the density is constant, this means that the volumetric flow is equal at both boundaries. Taking into consideration that the cross-sectional area of the pipe is equal at both boundaries, this means that the velocity is equal at both boundaries. The final two assumptions (6 and 7) were made in order to greatly simplify the energy balance equation. Assumption 6 is a common simplification of this energy balance [19] and

involves ignoring the contribution from turbulent fluctuations to the average velocity in the pipe, while assumption 7 is derived from a dimensional analysis of the E_v term [19]. These assumptions are listed in mathematical terms below:

Assumptions:

1. incompressible fluid ($E_c = 0$)

2. at zero elevation ($\phi_{tot}, \hat{\phi} = 0$)

3. no work done on system ($W_m = 0$)

4. constant density $\rho_1 = \rho_2$

5. $\Delta v = 0$ at plane 1 and plane 2

6. estimate E_v as $\frac{1}{2} < v^2 > e_v w$

7. estimate $\frac{<v^3>}{<v>}$ as v^2

Using assumptions 1-3, Eq. 3.1 reduces to:

$$\frac{d}{dt}(K_{tot}) = -\Delta(\frac{1}{2}\frac{< v^3 >}{< v >} + \frac{p}{\rho})w - E_v \qquad (3.2)$$

Using assumption 5, Eq. 3.2 yields:

$$\frac{d}{dt}(K_{tot}) = -\frac{\Delta p}{\rho}w - E_v \qquad (3.3)$$

35

Using assumption 6, $K_{tot} = \int \frac{1}{2}\rho v^2 dV$, and the chain rule $\frac{dv^2}{dt} = 2v\frac{dv}{dt}$, Eq. 3.3

yields:

$$\frac{d}{dt}\left(\int \frac{1}{2}\rho v^2 dV\right) = -\frac{\Delta p}{\rho}w - \frac{1}{2}v^2 e_v w \tag{3.4}$$

$$\frac{1}{2}\rho\left(\int dV\right)\frac{dv^2}{dt} = -\frac{\Delta p}{\rho}w - \frac{1}{2}v^2 e_v w \tag{3.5}$$

$$\rho v V \frac{dv}{dt} = -\frac{\Delta p}{\rho}w - \frac{1}{2}v^2 e_v w \tag{3.6}$$

Using the fact that $w = \rho v A_p$ and dividing through by v:

$$\rho V \frac{dv}{dt} = -\Delta p A_p - \frac{1}{2}v^2 e_v \rho A_p \tag{3.7}$$

where $\Delta p = p_1 - p_2 = P_{sys} = P_{retentate} = P_{feed}$ (in Pa; can be assumed to be equal to

the feed pressure if the pressure drop in the RO membrane modules is neglected, as is

the case in the following derivation). In future experiments, if the pressure drop in the

RO membrane modules is known, a more accurate value for P_{sys} can be used (exact

value from sensor measurements, or average between feed and retentate pressures).

Thus, Eq. 3.7 yields:

$$\rho V \frac{dv}{dt} = P_{sys} A_p - \frac{1}{2}v^2 e_v \rho A_p \tag{3.8}$$

which is essentially a momentum balance around control volume 2, and can be sim-

plified to yield:

$$\frac{dv}{dt} = \frac{P_{sys}A_p}{\rho V} - \frac{1}{2}\frac{A_p e_v v^2}{V} \tag{3.9}$$

Equation 3.9 relates the velocity downstream from the valve to the valve resistance. It can also be seen that the momentum balance results in a non-linear equation (non-linear with respect to v). This valve resistance value is a dimensionless quantity which is equal to zero for an absence of resistance (resulting in "infinite" flow), and goes to infinity as the valve becomes completely closed (resulting in zero flow).

In order to cast Eq. 3.9 in terms of only system parameters (e.g., system volume, membrane area, overall feed-side mass transfer coefficient, etc.) and state variables (e.g., retentate stream velocity), the P_{sys} term must be replaced by a function of the states. To do this, an overall mass balance of the system is utilized, represented by control volume 1 in Fig. 3.1.

Overall Mass Balance

The mass balance around control volume 1 can be represented as follows:

$$0 = [A_p^f v_f \rho] - [A_p^r v_r \rho + A_p^p v_p \rho] \tag{3.10}$$

where v_f is the feed stream velocity in m/s, v_r is the retentate stream velocity in m/s, v_f is the permeate stream velocity in m/s, A_p^f is the cross-sectional area of the feed line, A_p^r is the cross-sectional area of the retentate line, and A_p^p is the cross-sectional area of the permeate line (all cross-sectional areas in m^2). In the case of

the M3 system, the feed, retentate, and permeate lines have the same cross-sectional area (where flow measurement is taken), meaning that in this special case (utilized in Chapter 4, Chapter 5, and Appendix A), A_p values can be canceled out. In most systems, this is not the case, and the specific A_p values must be taken into account when conducting this derivation. Approximating the density of the water as being concentration invariant, Eq. 3.10 can be expressed as:

$$0 = A_p^f v_f - A_p^r v_r + A_p^p v_p \tag{3.11}$$

Using the algebraic relation relating permeate velocity to the transmembrane pressure, $v_p = k(P_{sys} - \Delta\pi)$ (again, using P_{sys} neglects pressure drop in the RO membrane modules; a more accurate average ΔP_m can also be used), Eq. 3.11 yields:

$$0 = v_f A_p^f - v_r A_p^r - k A_p^p (\Delta P_m - \Delta\pi) \tag{3.12}$$

where $k = \frac{A_m K_m}{\rho A_p^p}$ (or more commonly represented as $k = \frac{L_p A_m}{A_p^p}$) and $\Delta P_m = P_{sys} - P_p$ (P_p is taken to be the same as the pressure of the raw feed). Solving Eq. 3.12 for P_{sys} results in:

$$P_{sys} = \frac{\rho A_p^p}{A_m K_m}(A_p^f v_f - A_p^r v_r) + \Delta\pi \tag{3.13}$$

where K_m is the overall feed-side mass transfer coefficient (in m/s), A_m is the effective membrane area (in m^2), and L_p is the membrane permeability (in $m\ s^{-1}Pa^{-1}$).

Substituting Eq. 3.13 into the energy balance around the retentate valve (Eq. 3.9) yields:

$$\frac{dv}{dt} = \frac{A_p^p}{A_m K_m V}(A_p^f v_f - A_p^r v_r) + \frac{A_p^r}{\rho V}\Delta\pi - \frac{1}{2}\frac{A_p^r e_v v_r^2}{V} \qquad (3.14)$$

In Eq. 3.14, the $\Delta\pi$ term represents the osmotic pressure difference across the membrane (feed osmotic pressure minus permeate osmotic pressure at the membrane surfaces). As previously mentioned, this term is a function of the salt concentration on either side of the membrane, and the temperature of the solutions. Neglecting a temperature gradient between the feed and permeate sides of the membrane, the main factor remaining is the concentration of salt in the feed water at the membrane surface. Since the salt concentration is changing as the solution flows down the length of the membrane, a mathematical relation is needed to estimate the overall osmotic pressure. An approximate relation for the osmotic pressure is [56]:

$$\pi = 0.2641\left(\frac{Pa}{(ppm)(^\circ C)}\right)C_{effective}(T + 273) \qquad (3.15)$$

where the osmotic pressure, π, is in Pascals, $C_{effective}$ is in ppm, and T is in $^\circ C$. In a more comprehensive model, one can utilize multi-electrolyte thermodynamic calculations to determine π (and in turn, $\Delta\pi$) for complex systems.

Additionally, a value for the effective overall concentration, $C_{effective}$, in the membrane unit is required. This effective concentration can be a log-mean or arithmetic average of the solute concentration at the membrane surface on the feed side of the

39

RO membrane. For operation at low recovery, an average value may provide accurate results $(0.5(C_{ret} + C_{feed}))$, but for high recovery, this assumption fails to take into account the non-linearity of the salt concentration profile with respect to the distance from the entrance of the membrane. Work using the log-mean average [95] has also been conducted.

The concentration polarization phenomenon is not considered in the following derivation of $C_{effective}$, meaning that the actual salt concentration at the membrane surface will be higher than the theoretical model predicts. This will introduce a degree of error into the calculation of the osmotic pressure difference $(\Delta\pi)$. In the controller formulations presented in Chapters 4 and 5, this error can be partially accounted for by the lumped parameters (e.g., overall feed side mass transfer coefficient, and effective system volume) determined empirically through model verification on the experimental system.

For the control algorithms presented in Appendix A), an algebraic weighted relation for $C_{effective}$ is used:

$$C_{effective} = aC_{feed} + (1 - a)C_{retentate} \qquad (3.16)$$

where C_{feed} is the amount of total dissolved solids (TDS) in the feed (in ppm or mg/L), a is a dimensionless effective concentration weighting coefficient ($a = 0.5$ is used in Appendix A), and $C_{retentate}$ is the salt concentration in the retentate

(or concentrate) stream (in ppm or mg/L). As stated previously, the concentration polarization phenomenon is not taken into consideration in this algebraic weighted relation, which means that the actual osmotic pressure at the membrane surface (in the feed channel) will be higher than the value dictated by the model. For the purposes of energy optimal control (presented in Chapter 5), an equation directly relating the feed concentration to axially average osmotic pressure difference is used:

$$\Delta\pi = f_{os}C_{feed}\frac{ln(\frac{1}{1-Y})}{Y} \tag{3.17}$$

where f_{os} is a dimensionless empirically obtained constant ($f_{os} = 78.7$) [95], and Y is the overall system recovery:

$$Y = \frac{Q_p}{Q_f} \tag{3.18}$$

where Q_p is the permeate stream flow rate and Q_f is the feed stream flow rate.

This model is also advantageous because it captures the main characteristics of the M3 RO system. It is used for model development in the nonlinear model-based control algorithm (Chapter 4) and the energy-optimal optimization-based controller (Chapter 5).

In addition to the basic flow dynamics of the RO system, it is crucial to model the dynamics of the actuated valve. In the controllers presented in this work, it is necessary to relate the valve resistance to an actual valve position in order to

implement the control actions on the experimental system. To accurately model the valve dynamics and obtain practical constraints, the concept of valve C_v is used. The valve C_v is a standard relation of valve flow rate to pressure drop (across the valve) and is commonly used in valve sizing applications. The definition of C_v for a valve in a water system is presented in Eq. 3.19:

$$C_v = \frac{Q}{\sqrt{P_{sys}}} \tag{3.19}$$

where Q is the volumetric flow rate through the valve (in m^3/s), and P_{sys} is the system pressure (in Pa).

Assuming a pseudo-steady state condition (this assumption is not highly restrictive since valve operation time and transient behavior are relatively fast; properties of the actuated valve and operational speed are discussed further in Chapters 4, 5, and Appendix A), and using the simplified energy balance around the valve, (Eq. 3.9) Eq. 3.19 can be rearranged to yield:

$$C_v = \frac{1}{A_p \sqrt{\frac{1}{2}\rho e_v}} \tag{3.20}$$

Depending on the type of valve and its flow characteristics, it is assumed that the C_v value (and in turn, the e_v values from the model) can be related to the valve position (percentage open) through the following empirical logarithmic relation for the valve based on commercially available valve manuals [78]:

42

$$O_p = \mu \ln \left(\frac{1}{A_p \sqrt{\frac{1}{2}\rho e_v}} \right) + \phi \qquad (3.21)$$

where μ and ϕ are constants depending on the valve properties. This treatment of the valve characteristics allows for constraints based on valve actuator speed to be incorporated into the RO system model and into the controller calculations.

An additional model case is presented in Appendix A using two actuated valves (one as a bypass before the RO membranes in addition to the retentate valve) instead of the actuated retentate valve and variable frequency drive. It can be noted that the case using a variable frequency drive is preferable due to the reduction in energy cost for reducing the feed flow rate into the membrane units (see Chapter 5 for additional details on system energy usage).

3.2 Model Verification

Most of the parameters of the M3 system model, such as the membrane area (A_m), water density (ρ), pipe cross-sectional area (A_p), and system volume (V) have constant values which are inherent to the experimental system design (and current components installed). Another key model parameter, the overall feed-side mass transfer coefficient (K_m) was computed to match the model response to experimental step-test data. Specifically, K_m (defined as shown in Eq. 3.13) was computed using steady-state data from the experimental system by minimizing the difference between the model steady-

state and the experimental system steady-state for various step tests. Eq. 3.14 with the derivative term set to zero (steady-state) was solved for K_m. Experimental values for the constants A_p, A_m, V, ρ, the measured values (from system sensors) v_f, v_r, and the estimated values $\Delta\pi$ and e_v (from Eqs. 3.17 and 3.21, respectively) were used to calculate the value of K_m at a range of operating conditions. An initial average value for K_m (K_m^1) was determined through the experimental testing and used for the non-linear model-based control experiments in Chapter 4, while a slightly different value (K_m^2) was determined in a similar manner and used for the later experiments in Chapter 5. This difference in K_m can be attributed to changes of the spiral-wound membranes over time as well as the fact that the K_m value may change slightly at different operating conditions (it is noted that K_m is known to change with cross-flow velocity and permeate flux). The values of the model parameters can be found in Table 3.1.

The parameters for fitting the valve position to valve resistance value are also determined experimentally through similar steady-state tests using Eqs. 3.21, 3.20 and 3.19. The system is operated at several steady-states based on a range of valve open percentage (O_p); the pressure drop (across the valve) and flow rate data are then used to calculate C_v, which can be related to both the valve resistance value (e_v) and the open percentage of the valve (O_p) through fitting of the parameters μ and ϕ in Eq. 3.21. The values for μ and ϕ are determined for several ranges of valve positions. In Chapter 4, three separate equations (different μ and ϕ values)

Table 3.1: Process model parameters based on experimental system data.

ρ	$=$	1007	kg/m^3
V	$=$	0.6	m^3
A_p	$=$	0.000127	m^2
A_m	$=$	15.6	m^2
K_m^1	$=$	6.4×10^{-9}	s/m
K_m^2	$=$	9.7×10^{-9}	s/m
R	$=$	0.97	

are used to represent the full range of valve positions (Eqs. 4.2 - 4.4). In Chapter 5, five separate relations are introduced to represent the full range of valve positions (found in Table 5.1). Together, the model parameters determined by the steady-state step tests and the valve position-resistance relations form the basis of the connection between the theoretically derived analytical model and the actual M3 experimental RO system.

Chapter 4

Nonlinear Model-Based Control of Reverse Osmosis Water Desalination

4.1 Overview

Safe and reliable operation of a reverse osmosis desalination process in a field setting requires a control system capable of accounting for disturbances in feed water quality and the ability to quickly adjust to new desired operating conditions. Seasonal, monthly, or even daily changes in feed water quality can occur (depending on feed water source and location) and with a lack of appropriate system control, drive the RO system operation out of the range (system recovery, feed pressure, etc.) specified by the design. These deviations from the desired operating state can cause damage to system components through pressure fluctuations, or through the onset of RO membrane fouling/scaling (and subsequent membrane damage).

In order to mitigate the effects of changing feed water quality, a feedback lineariz-ing nonlinear model-based controller was developed and its effectiveness was demon-strated in conducting set-point transitions and rejecting feed salinity disturbances through application to the M3 experimental reverse osmosis desalination system. The dynamic nonlinear model presented in Chapter 3 is used to derive a nonlin-ear feedback linearizing controller to conduct set-point transitions of the retentate flow rate by adjusting an actuated retentate valve. Efficient operation of the reten-tate valve (and in turn, of the retentate stream flow rate/velocity) is integral to a reverse osmosis system because it (along with the feed pump speed) controls the clean water production rate and percentage of feed water which is discarded as the concen-trate stream. The nonlinear model-based controller is then implemented on the M3 experimental system where it is shown to possess excellent set-point tracking and dis-turbance rejection capabilities. The nonlinear controller is also shown to outperform a proportional-integral control system because it explicitly accounts for multivariable interactions.

4.2 RO System Model

Substituting Eq. 3.13 into the energy balance equation of Eq. 3.9 yields the following nonlinear ordinary differential equation (in the case of the M3 system where $A_p^f = A_p^r = A_p^p = A_p$) for the dynamics of the retentate stream velocity:

$$\frac{dv_r}{dt} = \frac{A_p^2}{A_m K_m V}(v_f - v_r) + \frac{A_p}{\rho V}\Delta\pi - \frac{1}{2}\frac{A_p e_{vr} v_r^2}{V} \qquad (4.1)$$

Using the above dynamic equation, various control techniques can be applied using the valve resistance value (e_{vr}, defined in Eq. 3.20) as the manipulated input. As the valve resistance approaches zero (in the limiting case which is not encountered in practice), the valve behaves as an open pipe; as the valve resistance approaches infinity, the valve behaves as a total obstruction and the flow velocity goes to zero [19]. It is also noted that K_m and $\Delta\pi$ are not independent of v_r and v_f, but for the purposes of the controller design presented in this chapter, it is assumed that K_m is constant and that $\Delta\pi$ can be calculated as shown in Eq. 3.17.

It is assumed that the C_v values (and in turn, the e_{vr} values) can be related to the valve position (percentage open) through the empirical logarithmic relation presented in Chapter 3 (Eq. 3.21).

The relation of valve position (O_p, in percent) as a function of valve resistance (e_{vr}) for the actuated retentate valve on the M3 system is shown in Fig. 4.1. It can be seen in Fig. 4.1 that as the valve position approaches zero (fully closed), the valve resistance values increase at a rapid rate; and as the valve approaches the fully-open position, the resistance values change slowly. The steady-state valve data relating O_p to e_{vr} obtained from the experimental system (see Chapter 3.2 for the analysis procedure) is also plotted in Fig. 4.1. The data does not fit the same logarithmic relation as the commercial theoretical valve curve [78]. With respect to

the accuracy of the sensor measurements of Fig. 4.1, the error bars corresponding to these measurements have been computed and are on the order of the markers used to denote the data points, and thus they have not been included in Fig. 4.1. The data was fit in three segments with curve fits following a similar form as the theoretical curve. The first curve fit was applied to valve resistance (e_{vr}) values of approximately 205 to 212 $(205 \leq e_{vr} < 212)$ and takes the form:

$$O_p = -844.28 \ln(e_{vr}) + 4592.1 \tag{4.2}$$

For e_{vr} values between 212 and 6200 $(212 \leq e_{vr} < 6200)$, O_p is computed by:

$$O_p = -20.473 \ln(e_{vr}) + 181.41 \tag{4.3}$$

while for e_{vr} values above 6200 $(e_{vr} \geq 6200)$, O_p is computed by:

$$O_p = -0.778 \ln(e_{vr}) + 9.476 \tag{4.4}$$

This treatment of the valve characteristics allows for conversion of the experimental values of O_p to values of e_{vr} in the model-based nonlinear control algorithm, and allows for values of e_{vr} generated by the control algorithm to be expressed in terms of
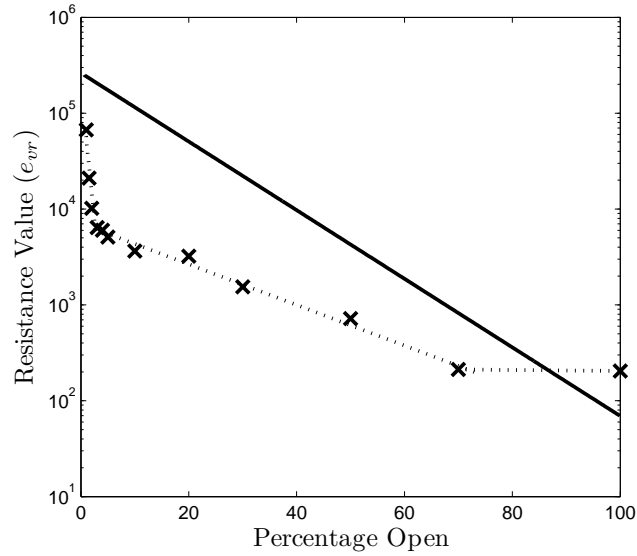
Figure 4.1: Correlation between retentate valve resistance value (e_{vr}) and retentate valve percentage open (O_p): commercial theoretical data (solid line), experimentally measured data for the retentate valve on the M3 system (x), and curve fittings to experimental data (dashed lines) using Eqs. 4.2-4.4. During steady-state experiments, system pressure set-points ranged from 85 to 270 psi, with retentate flow rates of 0.5 to 3 GPM.

values of O_p to be sent to the actuated valve on the experimental system. Capturing the nonlinearity present in the valve is extremely crucial when applying the control algorithms to the experimental system.

4.3 Control Algorithms

Two separate control loops are present in the control problem formulation (as shown in Figs. 4.2 and 4.3). The first loop regulates the system pressure by adjusting the variable frequency drive (VFD) speed directly (effectively changing the feed flow rate). This control loop will be termed "loop I". In each set of experiments presented below, a proportional-integral (PI) feedback controller is used to keep the system pressure (P_{sys}) at the set-point value (P_{sys}^{sp}) of 150 psi. This control algorithm takes the form:

$$S_{VFD} = K_f(P_{sys}^{sp} - P_{sys}) + \frac{K_f}{\tau_f} \int_0^{t_c} (P_{sys}^{sp} - P_{sys})dt \qquad (4.5)$$

where S_{VFD} is the control action (in volts, but in the case of the M3 system, the control voltage applied is equal to the pump flow rate in GPM) applied to the variable frequency drive (VFD speed), K_f is the proportional gain (in GPM/psi) and τ_f (in seconds) is the integral time constant.

The second control loop (termed "loop II") uses a nonlinear model-based controller (for the purposes of comparison, P and PI controllers are also used in loop II). The nonlinear controller utilizes the error between the retentate velocity and its corresponding set-point, but it also takes into account many additional system variables [29, 30, 27]. Specifically, the nonlinear model-based controller manipulates the

51

actuated retentate valve position by using measurements of the feed flow velocity (v_f), feed salinity (C_f), and retentate flow velocity (v_r), as demonstrated in Fig. 4.3, to bring the system to the desired operating condition (feed pressure and retentate stream flow rate). The nonlinear controller is designed following a feedback lineariza- tion approach [27]. To derive the controller formula, the following linear, first-order response in the closed-loop system between v_r and v_r^{sp} is requested:

$$\frac{dv_r}{dt} = \frac{1}{\gamma}(v_r^{sp} - v_r) \tag{4.6}$$

where γ is a variable parameter to adjust the magnitude of the response (in seconds). It is noted that a first-order response is requested because the relative degree between v_r and e_{vr} is one, as shown in Eq. 4.1 [27]. Using this approach, the following formula is obtained for the nonlinear controller:

$$e_{vr} = \frac{\frac{1}{\gamma}(v_r^{sp} - v_r) - \frac{A_p^2}{A_m K_m V}(v_f - v_r) - \frac{A_p \delta(T+273)}{\rho V}C_{eff}}{\frac{-A_p}{2V}(v_r^2)} \tag{4.7}$$

To achieve offset-less response, integral action is added to the controller in Eq. 4.7 and the resulting controller takes the form:

$$e_{vr} = \frac{\frac{1}{\gamma}(v_r^{sp} - v_r) + \frac{1}{\tau_{NL}} \int_0^{t_c} (v_r^{sp} - v_r)dt}{\frac{-A_p}{2V}(v_r^2)} +$$

$$\frac{-\frac{A_p^2}{A_m K_m V}(v_f - v_r) - \frac{A_p \delta(T+273)}{\rho V} C_{eff}}{\frac{-A_p}{2V}(v_r^2)}$$

(4.8)

As a baseline, the performance of the nonlinear controller is compared to a traditional form of control (P or PI, depending on the form of the nonlinear controller). Loop II, using P or PI control, uses the retentate (or concentrate) stream flow velocity to manipulate the actuated valve in order to regulate the retentate stream velocity/flow rate. Under P or PI control, the control system for loop II takes the form(s) (under P control):

$$O_p = K_r(Q_r^{sp} - Q_r) \tag{4.9}$$

or (under PI control):

$$O_p = K_r(Q_r^{sp} - Q_r) + \frac{K_r}{\tau_r} \int_0^{t_c} (Q_r^{sp} - Q_r)dt \tag{4.10}$$

where O_p is the valve position (in terms of % open), Q_r is the retentate stream volumetric flow rate and Q_r^{sp} is the retentate stream flow rate set-point (both given in GPM). In this case, the proportional gain, K_r is given in terms of %/GPM and the integral time constant τ_r is given in seconds.

In each set of experiments, the performance of the nonlinear controller implemented on the experimental system was compared to the performance of the nonlinear controller implemented on the process model and to the performance of a proportional or proportional-integral controller implemented on the experimental system. The control algorithms were programmed into the data acquisition and control software to operate in real-time with a sampling time of 0.1 seconds. Additionally, the actuated retentate valve is powered by an electric motor with a maximum operating speed which must be taken into account when attempting to simulate the nonlinear controller action. From testing on the experimental M3 system, it was found that the actuated valve could travel its entire range in approximately 45 seconds; this provides an important constraint on the speed of valve opening/closing in the simulations (and in the experimental M3 system) of the form:

$$\left| \frac{dO_p}{dt} \right| \leq 2.22 \frac{\%}{s} \tag{4.11}$$

In order to derive the constraint of Eq. 4.11, it is assumed that the valve speed is independent of valve position (valve always turns at maximum speed). This is a physical constraint which is intrinsically accounted for in the closed-loop experimental results and is programmed into the nonlinear model-based controller simulation as well (to facilitate comparison). Additionally, when using the experimental M3 system, the valve position is not allowed to fall under 1%, and any valve position values sent to
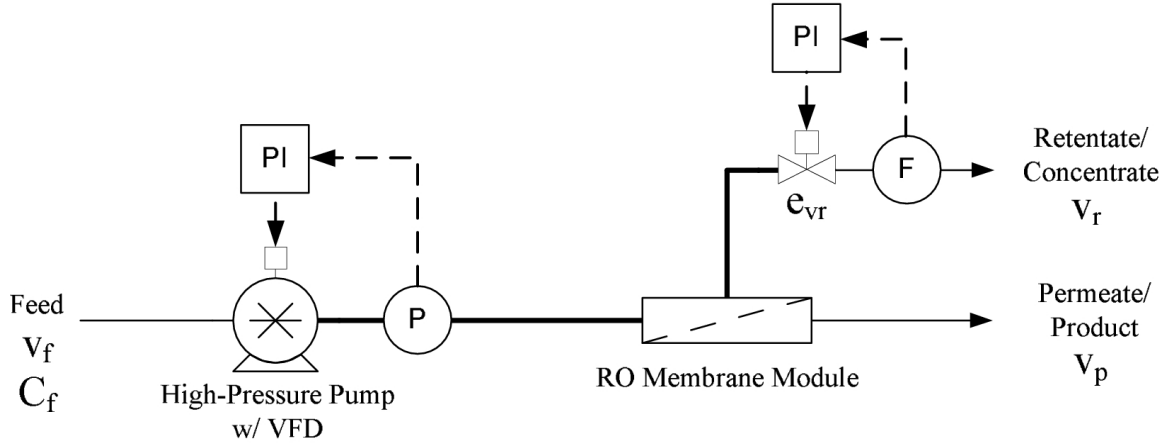
Figure 4.2: Reverse osmosis system under two proportional-integral control systems: squares indicate proportional-integral controllers (PI), circles indicate measurement sensors (pressure (P), flow (F)).

the valve above 100% are translated to the maximum value of 100% open. The lower constraint ($> 1\%$) is enforced so that the system pressure will not rise too rapidly and exceed component pressure limitations. A constraint on the variable frequency drive is also placed to avoid pressure spikes (a maximum VFD speed of 4.5/10 is used in order to operate below 400 psi). In the experiments presented in this work, the actuators do not reach these constraints.
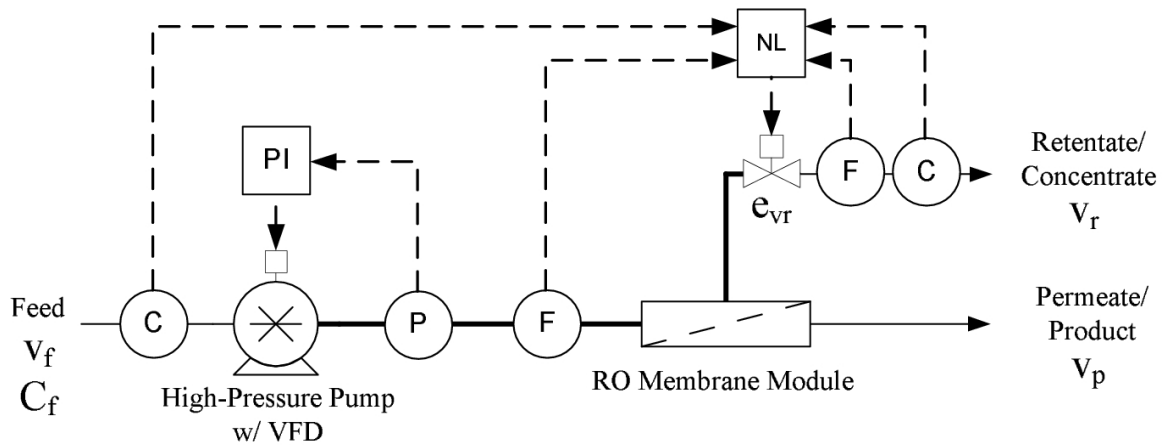
Figure 4.3: Reverse osmosis system under proportional-integral control adjusting the VFD speed and nonlinear control adjusting retentate valve position: squares indicate proportional-integral control (PI) and nonlinear control (NL), circles indicate measurement sensors (conductivity (C), pressure (P), flow (F)).

4.4 Experimental Closed-Loop Results

Using the controller formulation detailed above, it was desired to examine the performance of the nonlinear model-based controller when compared to classical proportional and proportional-integral controllers for large changes in system set-points (retentate stream flow rate and system pressure). The following sets of experiments also demonstrate the ability of the nonlinear model-based controller to maintain a system operating state in a situation where a large change in feed water salinity is observed. In the control experiments presented in this chapter, the experimental system was turned on and the PI loop controlling the variable frequency drive (loop I) was activated to bring the system pressure to a set-point of $P_{sys} = 150$ psi. The retentate flow rate was set to 1.5 gallons per minute (gpm). After the system had reached a steady-state (with respect to system pressure and flow rates), loop II was activated to manipulate the retentate valve. All data taken from the experimental system was averaged using a 19 point (approximately two seconds of data) moving average to remove some of the measurement noise in order to increase the clarity of the presented data. The closed-loop response observed for the nonlinear controller applied to the dynamic process model is used as a baseline for comparison of controller performance, as well as to determine an approximate range of controller tunings for the experimental system.

4.4.1 Set-point Change Experiments

In the first set of experiments, the retentate flow rate set-point (interchangeable with retentate velocity set-point, $Q_r^{sp} = v_r^{sp} A_p$) was changed from 1.5 gpm to 3 gpm, while loop I was maintained at a pressure set-point of 150 psi. In these experiments, the closed-loop performance of the nonlinear controller (without integral action) implemented on the experimental system was first evaluated against the performance of the simulated nonlinear controller (using the derived process model) and the performance of an experimentally implemented proportional (P) controller. The feed salt concentration for the first set of experiments was approximately 5400 ppm of sodium chloride (NaCl). The tuning parameters for the controllers can be found in Tables 4.1 and 4.2. Both loops were tuned (in the simulation and the experimental system) to achieve a slightly under-damped closed loop output response (i.e., fastest approach to the steady-state with minimal oscillatory output response) [68, 77].

Table 4.1: Loop I PI controller tuning parameters for first set of experiments.

$$
\begin{aligned}
K_f &= 0.01 & GPM/psi \\
\tau_f &= 0.1 & s \\
K_f^{sim} &= 0.01 & GPM/psi \\
\tau_f^{sim} &= 0.1 & s
\end{aligned}
$$

The results of the experiments were recorded using the data acquisition and con-

Table 4.2: Loop II controller tuning parameters for first set of experiments (parameters for both P and nonlinear controllers).

$$
\begin{array}{rcll}
K_r & = & 1 & \%/GPM \\
\gamma & = & 0.6 & s \\
\gamma^{sim} & = & 0.6 & s
\end{array}
$$
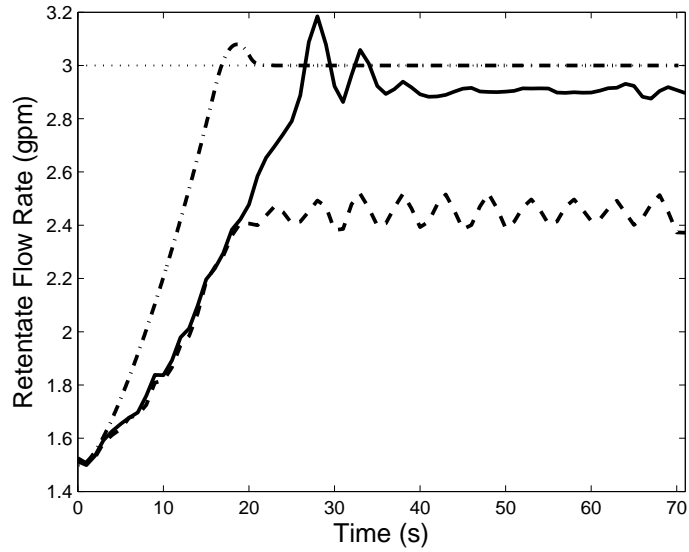
Figure 4.4: Profiles of retentate flow rate (Q_r) with respect to time for retentate flow rate set-point transition from 1.5 to 3 gpm under proportional control (dashed line), nonlinear model-based control (solid line) and nonlinear model-based control implemented via simulation on the process model (dash-dotted line). The horizontal dotted line denotes the retentate flow rate set-point $(Q_r^{sp} = 3 \text{ gpm})$.
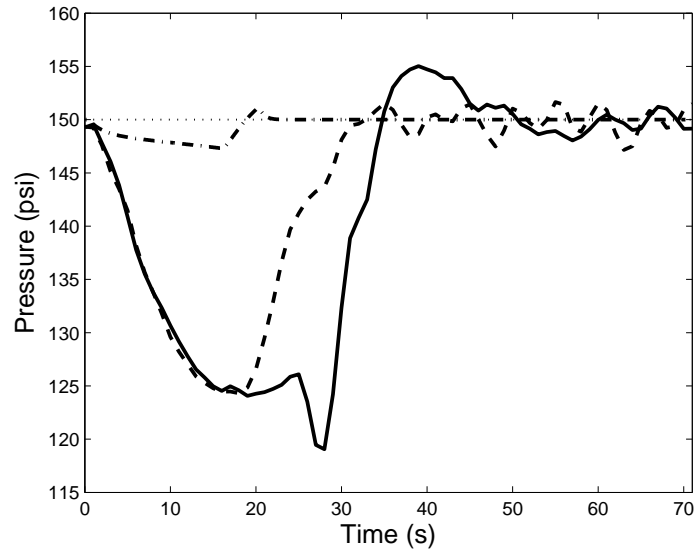
Figure 4.5: Profiles of system pressure (P_{sys}) with respect to time for retentate flow rate set-point transition from 1.5 to 3 gpm under proportional control (dashed line), nonlinear model-based control (solid line) and nonlinear model-based control implemented via simulation on the process model (dash-dotted line). The horizontal dotted line denotes the system pressure set-point ($P_{sys}^{sp} = 150$ psi).
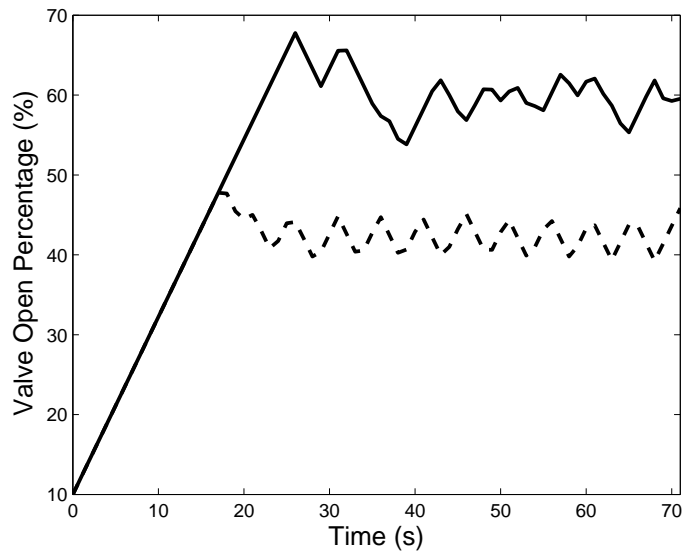
Figure 4.6: Profiles of valve open percentage (O_p) with respect to time for retentate flow rate set-point transition from 1.5 to 3 gpm under proportional control (dashed line) and nonlinear model-based control (solid line).
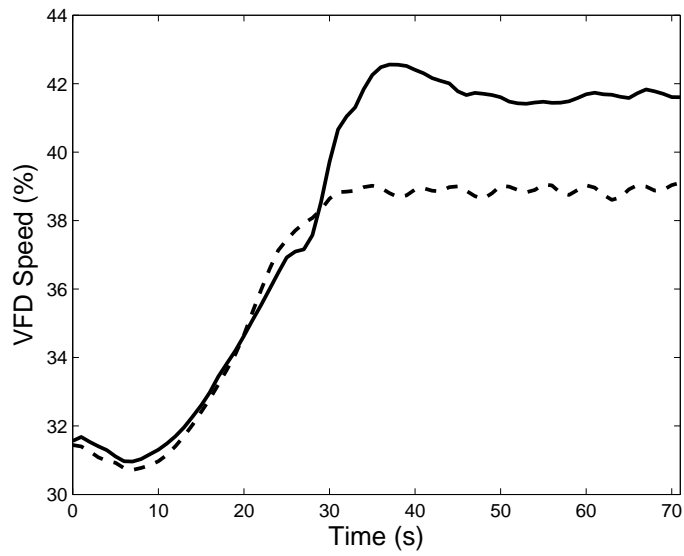
Figure 4.7: Profiles of variable frequency drive speed with respect to time for retentate flow rate set-point transition from 1.5 to 3 gpm under proportional control (dashed line) and nonlinear model-based control (solid line).

trol interface and are plotted in Figs. 4.4 - 4.7. In Fig. 4.4, it can be seen that the simulated model-based nonlinear controller yields a closed-loop response that converges the fastest to the steady-state and achieves the smallest offset of any of the controllers. The above results are expected since the simulated controller is not subject to any type of plant-model mismatch or any measurement noise. The proportional controller implemented on the experimental M3 system shows a significant offset (approximately 20% from the set-point) and also demonstrates sustained oscillations after it levels off. These fairly large oscillations in retentate flow rate (caused by oscillations in the valve position) also lead to system pressure oscillations. These oscillations in the system pressure lead to oscillations in the pump speed (from loop I) and can be detrimental to the operating life of the pumps. The nonlinear model-based controller (when applied to the experimental system) is shown to have a much smaller offset than the proportional controller in the same set-point tracking experiment (approximately 3-4% from the set-point). Brief oscillations are observed as the controller slightly overshoots the set-point, but these oscillations decay and the system quickly stabilizes at the new set-point. The smoothness of the closed-loop response under nonlinear control is due to the fact that it takes into account the action of loop I in the computation of the control action, while the proportional controller neglects the highly coupled nature of the two control loops. As the valve opens to allow for more flow through the retentate line (as dictated by the proportional controller), the system pressure decreases, causing loop I to increase the feed flow rate to maintain the system

pressure at the set-point. As a result of this, the retentate flow rate increases, forcing the proportional controller acting on the retentate valve to begin closing the valve. This interplay between the controllers causes the observed oscillations and can result in an increased time to reach the desired set-point. In Fig. 4.5, it can be seen that the simulated nonlinear controller demonstrates the smallest deviation from the system pressure set-point (again, this result is expected since the simulation is not subject to plant-model mismatch or measurement noise). Comparing the performance of the proportional controller and of the nonlinear controller when they are applied to the experimental system, it can be seen that the system response under the nonlinear controller deviates slightly more than under the proportional controller (approximately 5 psi in each direction), but the closed-loop pressure under the proportional controller demonstrates sustained periodic oscillations.

When examining the valve movement as depicted in Fig. 4.6, it can be seen that in both cases (using the proportional controller and the nonlinear model-based controller), the initial valve position change is identical due to the constraint on the maximum rate of valve opening/closing (Eq. 4.11). The nonlinear controller opens the valve to a greater extent (around 60%) than the proportional (P) controller (around 42%), leading to a smaller offset (larger retentate flow rate with higher valve position). The variable frequency drive input profiles in Fig. 4.7 show that due to the larger valve position value requested by the nonlinear controller, the variable frequency drive must speed up slightly (39% of maximum vs. 42% of maximum for

system under proportional control) to achieve the system pressure set-point of 150 psi. When comparing the performance of the nonlinear controller to the performance of the proportional controller for retentate set-point changes (in Fig. 4.4), it can be seen that even though the nonlinear controller causes slightly greater fluctuations in system pressure (additional deviation of 5 psi from set-point during transition), the retentate flow rate offset from the set-point is much smaller (0.6 gpm offset for proportional control vs. 0.1 gpm offset for nonlinear model-based control), and the oscillations are minimized.

In the second set of experiments, the retentate flow rate set-point was changed from an initial value of 1.5 gpm to a new value of 0.8 gpm, while the VFD control loop was again maintained at a pressure set-point of 150 psi. In this set of experiments, the performance of the nonlinear controller with integral term was evaluated against the performance of a proportional-integral (PI) controller (both of these controllers are implemented experimentally), and the performance of the nonlinear controller with integral action applied to the dynamic process model via simulations. The feed salt concentration for these experiments was approximately 8200 ppm of NaCl. The tuning parameters for the controllers in this set of experiments can be found in Tables 4.3 and 4.4.

The results for the second set of experiments are plotted in Figs. 4.8 - 4.11. Results for an alternate PI controller tuning are also presented in Figs. 4.12 - 4.15. This alternate tuning was presented to demonstrate the limited applicability of PI

Table 4.3: Loop I PI controller tuning parameters for second set of experiments.

K_f	$=$	0.01	GPM/psi
τ_f	$=$	0.1	s
K_f^{sim}	$=$	0.0091	GPM/psi
τ_f^{sim}	$=$	0.1	s

Table 4.4: Loop II controller tuning parameters for second set of experiments (parameters for both PI and nonlinear controllers).

K_r	$=$	1	$\%/GPM$
τ_r	$=$	(see figure captions)	s
γ	$=$	0.6	s
τ_{NL}	$=$	10	s
γ^{sim}	$=$	0.6	s
τ_{NL}^{sim}	$=$	10	s
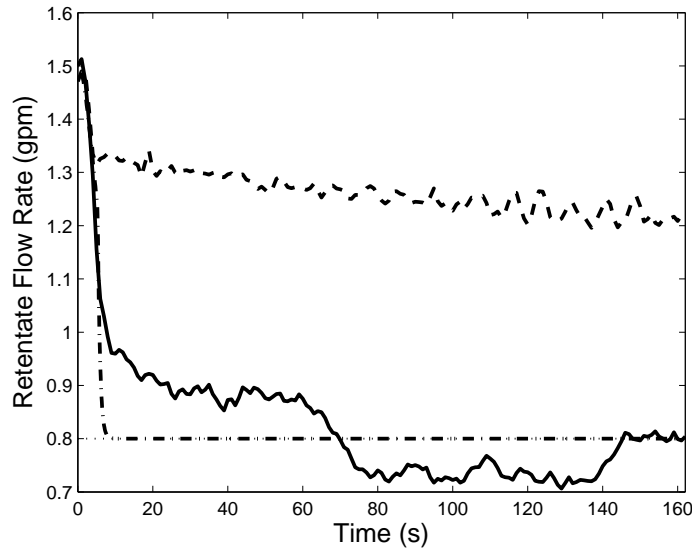
Figure 4.8: Profiles of retentate flow rate (Q_r) with respect to time for retentate flow rate set-point transition from 1.5 to 0.8 gpm under proportional-integral control (dashed line) with $\tau_r = 5\ s$, nonlinear model-based control with integral action (solid line) and nonlinear model-based control with integral action implemented via simulation on the process model (dash-dotted line). The horizontal dotted line denotes the retentate flow rate set-point $(Q_r^{sp} = 0.8$ gpm).
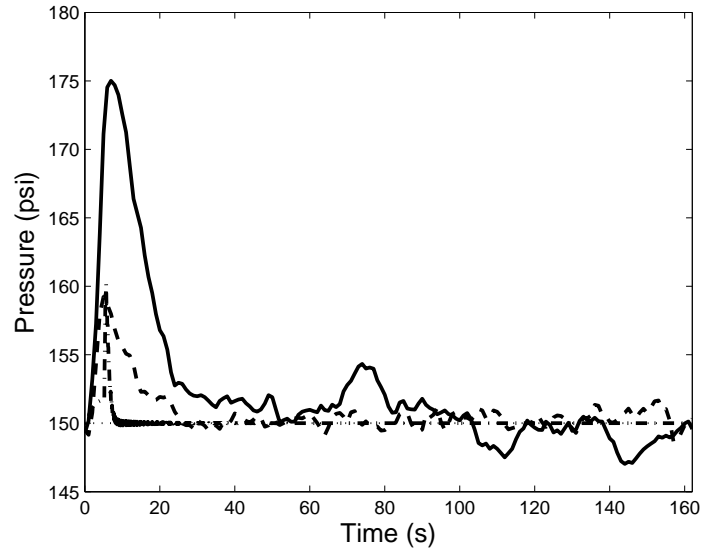
Figure 4.9: Profiles of system pressure (P_{sys}) with respect to time for retentate flow rate set-point transition from 1.5 to 0.8 gpm under proportional-integral control (dashed line) with $\tau_r = 5$ s, nonlinear model-based control with integral action (solid line) and nonlinear model-based control with integral action implemented via simulation on the process model (dash-dotted line). The horizontal dotted line denotes the system pressure set-point ($P_{sys}^{sp} = 150$ psi).
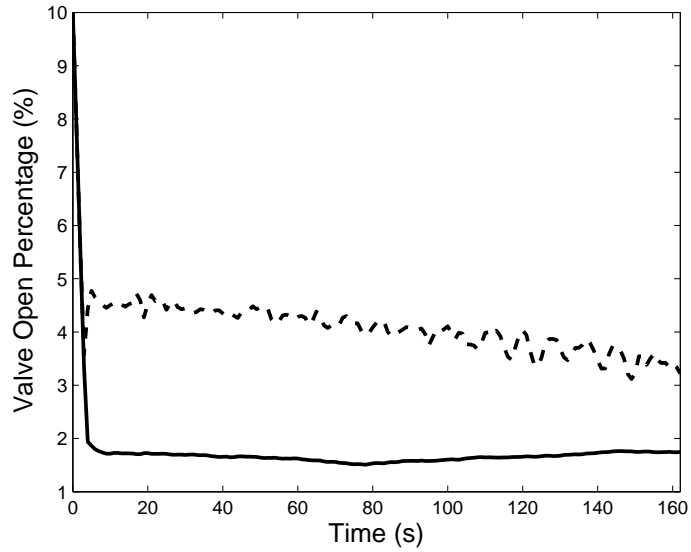
Figure 4.10: Profiles of valve open percentage (O_p) with respect to time for retentate flow rate set-point transition from 1.5 to 0.8 gpm under proportional-integral control (dashed line) with $\tau_r = 5$ s and nonlinear model-based control with integral action (solid line).
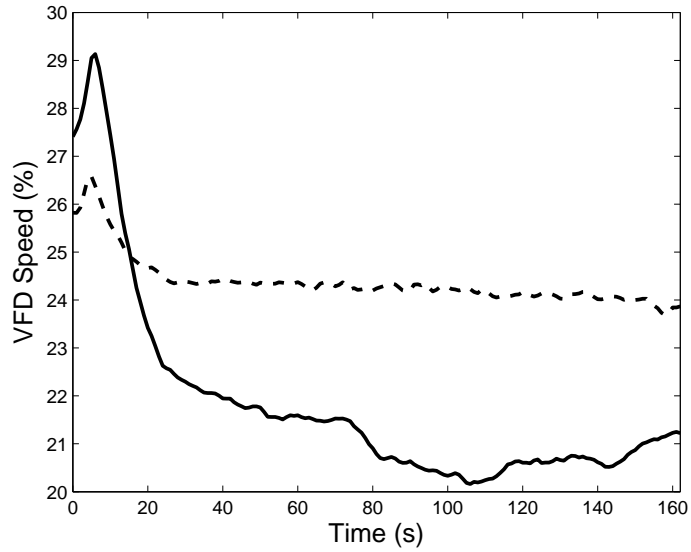
Figure 4.11: Profiles of variable frequency drive speed with respect to time for re-tentate flow rate set-point transition from 1.5 to 0.8 gpm under proportional-integral control (dashed line) with $\tau_r = 5\ s$ and nonlinear model-based control with integral action (solid line).
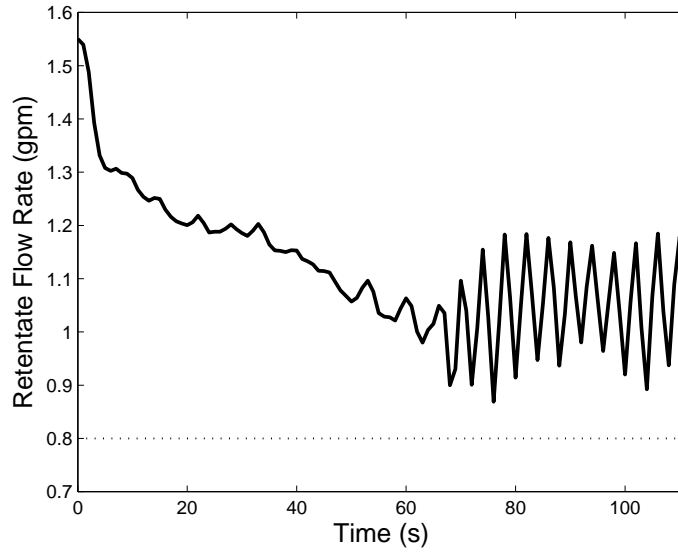
Figure 4.12: Profile of retentate flow rate with respect to time for retentate flow rate set-point transition from 1.5 to 0.8 gpm under proportional-integral control ($\tau_r = 0.7\ s$). The horizontal dotted line denotes the retentate flow rate set-point ($Q_r^{sp} = 0.8$ gpm).
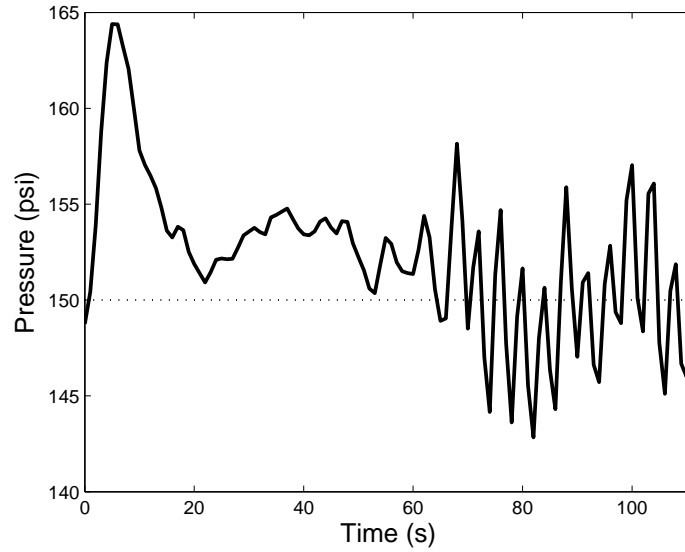
Figure 4.13: Profile of system pressure (P_{sys}) with respect to time for retentate flow rate set-point transition from 1.5 to 0.8 gpm under proportional-integral control ($\tau_r = 0.7\ s$). The horizontal dotted line denotes the system pressure set-point ($P_{sys}^{sp} = 150$ psi).
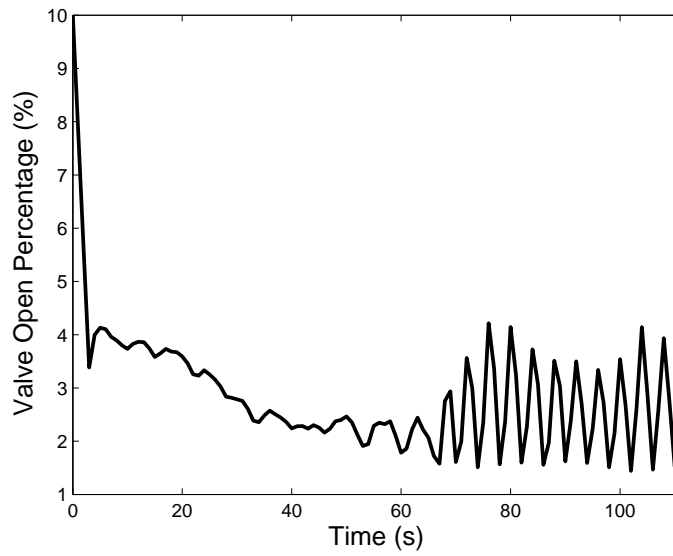
Figure 4.14: Profile of valve open percentage (O_p) with respect to time for retentate flow rate set-point transition from 1.5 to 0.8 gpm under proportional-integral control ($\tau_r = 0.7\ s$).
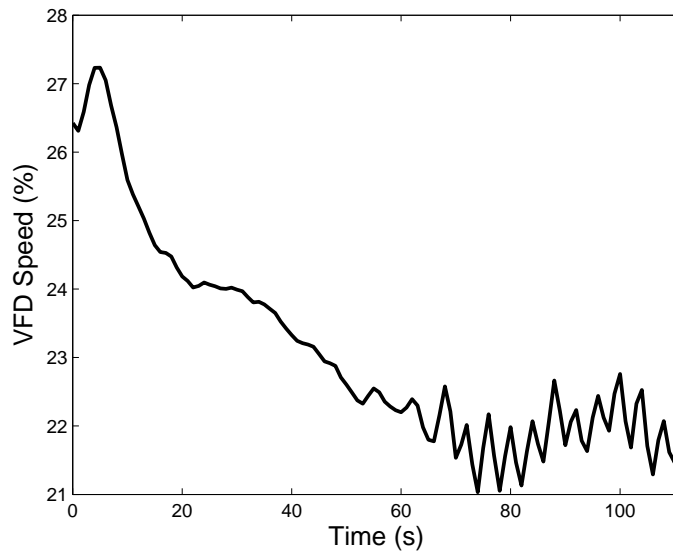
Figure 4.15: Profile of variable frequency drive speed with respect to time for retentate flow rate set-point transition from 1.5 to 0.8 gpm under proportional-integral control ($\tau_r = 0.7 \ s$).

control to this system under a range of integral time constants.

In Fig. 4.8, it can be seen that all of the closed-loop results (simulated and experimental) decrease at the same rate initially (due to the valve opening/closing rate constraint). As expected, the simulated nonlinear model-based controller with integral term immediately converges to the set-point with no offset since it is not subject to any plant-model mismatch or measurement noise. As it is evident in Table 4.3, the integral time constant for the simulated controller is slightly different ($\tau_f = 0.01$, $\tau_f^{sim} = 0.0091$). The simulations where the nonlinear controller was applied to the process model were used to find an approximate range of controller parameters, but these values were implemented on the experimental system and changed through systematic testing to achieve better closed-loop performance in the presence of plant-model mismatch. The speed of the closed-loop response under the nonlinear controller applied to the experimental system is slower in terms of convergence to the set-point than the one in the simulated case and the retentate flow rate reaches the set-point in about 145 seconds. The proportional-integral (PI) controller with $\tau_r = 5$ leads to an extremely slow convergence to the set-point (on the order of 10 minutes). In Fig. 4.12, a PI controller with the same gain K_r but with $\tau_r = 0.7$ is used to demonstrate the other extreme. This PI controller leads to a closed-loop response that converges to the set-point at a faster rate than the PI controller with $\tau_r = 5$, but as it can be seen, it results in significant oscillations around the set-point due to the coupling between the two control loops. These oscillations cause fluctuations in the feed flow

rate (due to the VFD control loop) and could damage the feed pumps and cause fatigue on system components.

Similar results to those presented for the case with proportional control are evident in Fig. 4.9 for the application of the nonlinear controller to the experimental system. The nonlinear model-based controller causes the largest deviation from the pressure set-point due to the speed at which it converges to the set-point. It can be seen that the PI controller has little effect on the system pressure (deviation of 10 psi) when compared to the effect of the nonlinear model-based controller (deviation of 25 psi) because the convergence to the retentate flow rate set-point is much slower. As the valve closes, it causes the system pressure to rise, forcing loop I to take action in order to keep the system pressure at the set-point. Slower valve actions allow more time for loop I to act and keep the system pressure at the set-point, such as in the case of the PI control with $\tau_r = 5$. The case of the PI controller where $\tau_r = 0.7$ (Fig. 4.13) shows that the initial pressure spike is much larger than in the case where $\tau_r = 5$, but smaller than the one under the nonlinear controller.

A similar explanation applies in Fig. 4.10, when looking at the valve positions for the various controllers. Specifically, it is observed that the valve position in the case of PI control (for both cases: $\tau_r = 5$ and $\tau_r = 0.7$ in Fig. 4.14) is much more erratic and results in system pressure oscillations of approximately 15 psi. From the results of the second set of experiments, it is again observed that the nonlinear controller achieves quick set-point transition with no offset (due to the addition of the integral

term), while minimizing oscillations more effectively than the proportional-integral (PI) controllers. The variable frequency drive speed profiles show a similar trend as in the first set of experiments (Fig 4.11); since the valve position value is smaller for the nonlinear controller, the variable frequency drive slows down in order to maintain the set-point system pressure.

4.4.2 Feed Quality Disturbance Experiments

As described in the introduction, the variability of feed water quality is also an important issue when designing controllers for reverse osmosis systems. Although the time scales of feed water quality variation are usually quite large (hours, days, or even weeks), the third set of experiments was designed to test the robustness of the controller when presented with a large change in feed water quality on a smaller time scale (on the order of 10-30 seconds). The controller parameters used in these experiments are shown in Tables 4.5 and 4.6. As in the first two experiments, the retentate flow rate set-point was 1.5 gpm, and the system pressure set-point was 150 psi. Two relatively large pulses of sodium chloride (NaCl) were added to the system while it was operating under nonlinear model-based control with integral action. The feed concentration over the course of the experiment can be seen in Fig. 4.16. The first pulse ("pulse" denotes an addition of salt to the feed tank over the period of 1-2 seconds) of salt was added at approximately 90 seconds, bringing the feed conductivity from 5500 μS up to a final value of near 7000 μS. The second pulse was added after

the feed salinity stabilized and brought the feed conductivity up to a final value of about 8000 μS (after mixing). The effects of these pulses on valve position, retentate flow rate, and system pressure can be seen in Figs. 4.17 - 4.20.

Table 4.5: Loop I PI controller tuning parameters for feed disturbance experiments.

$$
\begin{aligned}
K_f &= 0.01 \quad GPM/psi \\
\tau_f &= 0.1 \quad s
\end{aligned}
$$

Table 4.6: Loop II controller tuning parameters for feed disturbance experiments.

$$
\begin{aligned}
\gamma &= 0.6 \quad s \\
\tau_{NL} &= 10 \quad s
\end{aligned}
$$

In Fig. 4.17, it can be seen that through all of the feed salt concentration changes, the nonlinear model-based controller keeps the retentate flow rate within 2-3% of the set-point value of 1.5 gpm. Fig. 4.18 also demonstrates that the control system is able to keep the system pressure within similar bounds from the set-point value of 150 psi. When examining the control action in Fig. 4.19, it can be seen that the valve slowly closes. Specifically, as the feed concentration increases, the osmotic pressure resisting flow through the membrane also increases. If the system pressure is kept constant, this forces a greater percentage of the water to stay in the concentrate stream (lower driving force through the membrane due to the osmotic pressure increase). Since the retentate stream is controlled at a set-point of 1.5 gpm, the controller closes the
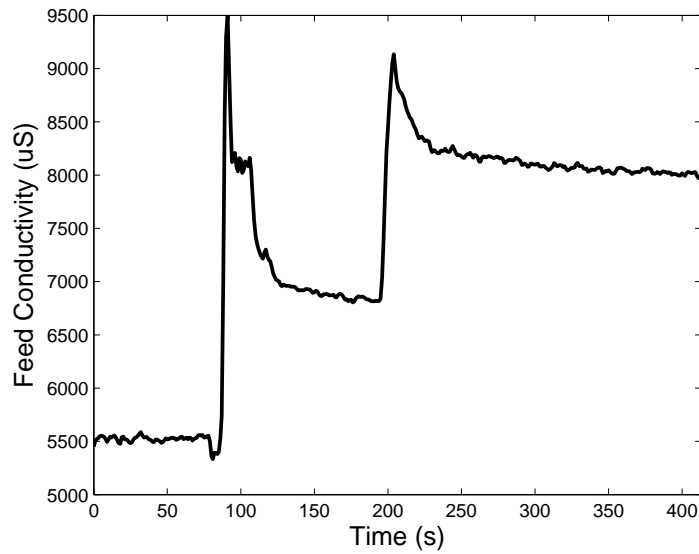
Figure 4.16: Profile of feed conductivity with respect to time; disturbance rejection experiment under nonlinear control with integral action.

actuated valve in order to mitigate this effect. When examining the variable frequency drive speed control action, it can be seen that the VFD speed slowly decreases as the feed conductivity increases. This trend makes sense because the water entering the membrane units is facing more resistance due to increased osmotic pressure across the membranes. This increase in osmotic pressure raises the system pressure for a fixed feed flow rate; therefore, the VFDs must slow down the feed flow rate in order to maintain the set-point system pressure. Again, the nonlinear controller is shown to perform very well in the presence of feed salt concentration variability. It is also noted that the size of the moving average window (1.9 s) is small enough to reduce a significant amount of the measurement noise, but it is also small when compared to

79
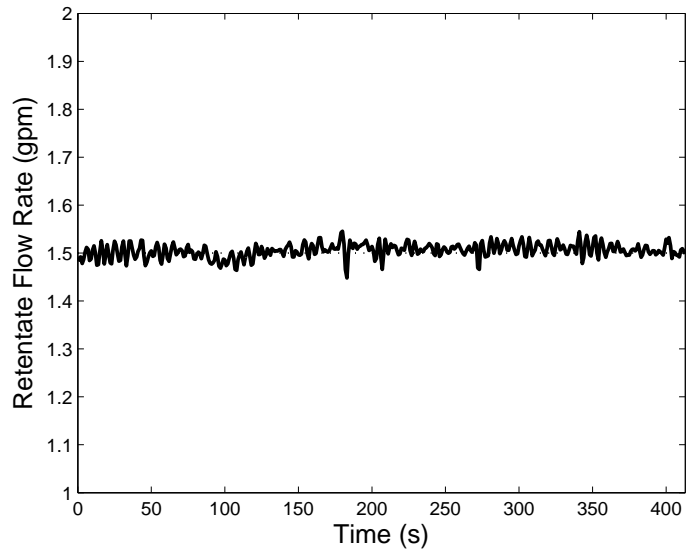
Figure 4.17: Profile of retentate flow rate with respect to time; disturbance rejection experiment under nonlinear control with integral action (solid line). The horizontal dotted line is the retentate flow rate set-point (1.5 gpm). Feed pressure was maintained at the set-point of 150 psi.
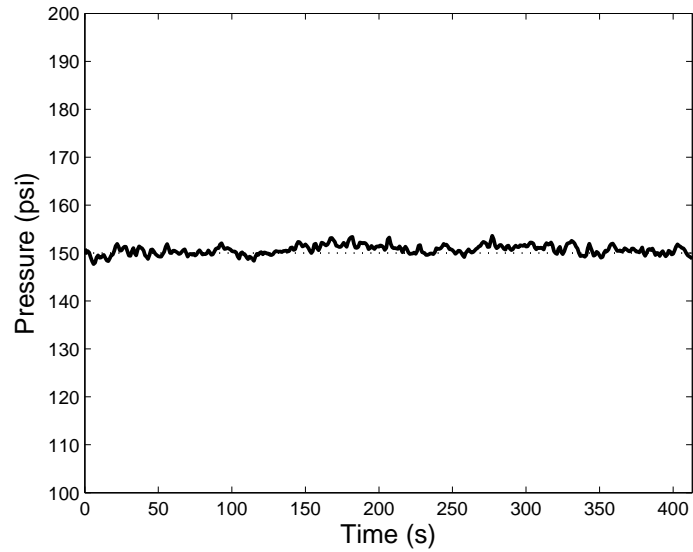
Figure 4.18: Profile of system pressure (P_{sys}) with respect to time; disturbance rejection experiment under nonlinear control with integral action (solid line). The horizontal dotted line is the pressure set-point ($P_{sys}^{sp} = 150$ psi). The retentate flow rate was maintained at the set-point of 1.5 gpm.
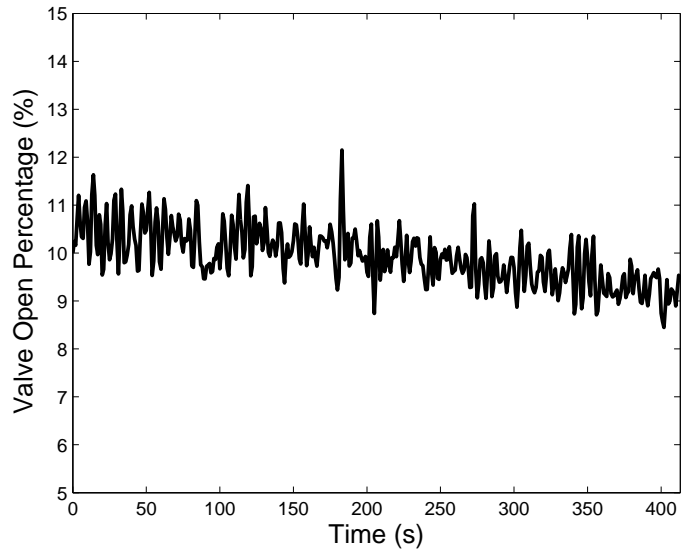
Figure 4.19: Profile of valve open percentage (O_p) with respect to time; disturbance rejection experiment under nonlinear control with integral action. The system pressure was maintained at the set-point of 150 psi and the retentate flow rate was maintained at the set-point of 1.5 gpm.
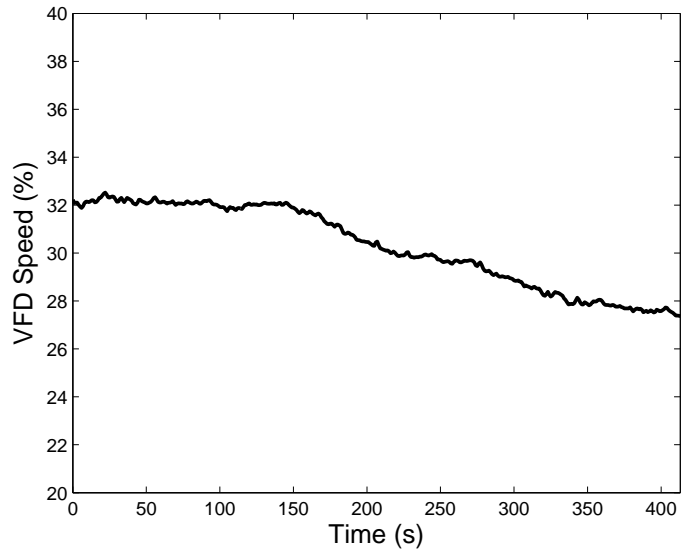
Figure 4.20: Profile of variable frequency drive speed with respect to time; disturbance rejection experiment under nonlinear control with integral action. The system pressure was maintained at the set-point of 150 psi and the retentate flow rate was maintained at the set-point of 1.5 gpm.

the transient system response time. It can also be seen from the disturbance rejection experiment that when a large change in salinity occurs, the system reaches steady-state again on the order of 100 seconds. In an industrial RO system, the feed water salinity will change much slower, on the order of minutes, hours or even days. From the experimental results, it is shown that the nonlinear model-based controller is able to account for large changes in feed salinity and maintain system operation at the desired operating pressure and retentate flow rate.

4.5 Conclusions

In this chapter, a nonlinear model-based control strategy was developed and experimentally implemented on a reverse osmosis membrane water desalination system. First a dynamic fundamental model describing the reverse osmosis desalination system was derived; the parameters of this model were then computed using step test data from UCLA's M3 experimental reverse osmosis desalination system. Specifically, correlations were derived relating the actuator position to model parameters, and the remaining model parameters were computed based on the experimental data. A nonlinear model-based control algorithm was then designed based on the constructed process model and the principle of feedback linearization. This nonlinear controller was implemented in order to maintain system operation at the desired pressure and retentate flow rate operating point by manipulating the actuated retentate valve along with a proportional-integral controller employed to manipulate the variable frequency

drive speed adjusting the feed flow rate. The performance of the nonlinear controller was compared to the performance of proportional and proportional-integral control algorithms, as well as benchmarked against the simulated nonlinear model-based controller during retentate flow rate set-point transitions. It was demonstrated that the nonlinear controller is much better suited to deal with the highly coupled RO system dynamics during set-point transitions and was shown to outperform the traditional control schemes. The model-based nonlinear controller also performed well when the experimental reverse osmosis system was subjected to a series of large step changes in feed salt concentration.

Chapter 5

Energy Optimal Control of Reverse Osmosis Water Desalination

5.1 Overview

Subsequent to the design and implementation of the control algorithms to control the steady-state transition (transient) properties of the experimental RO system, and to mitigate the effects of disturbances of feed water quality (both accomplished in Chapter 4), it was desired to optimize the steady-state RO system operation with respect to energy usage under varying feed water conditions. In a typical seawater RO plant, the cost of energy can approach 45% of the total permeate production cost due to the fact that the system operation can require very high feed pressures (800-1000 psi) in order to achieve a desired permeate production rate [61, 22, 92]. Several efforts have been made in order to decrease the energy required by a reverse osmosis desalination system; these include work in increasing membrane permeability leading to lower required transmembrane pressure [91, 98], optimization of RO module and

system configuration [88, 95, 100, 62], and the use of energy recovery devices [89]. In the area of energy optimization, it has been recently shown that the specific energy consumption, or SEC (energy cost per volume of permeate water produced), is a useful metric to quantify reverse osmosis water desalination system energy usage. Within the SEC framework, the issues of unit cost optimization with respect to water recovery, energy recovery, system efficiency, feed/permeate flow rate, membrane module topology [95, 97, 99], and optimization of the transmembrane pressure subject to feed salinity fluctuations [96] have been studied. However, experimental verification of the theoretically computed energy optimal operating points was not carried out in the aforementioned works [95, 97, 99, 96].

In this chapter, feedback control is integrated with an SEC-based energy optimization algorithm in order to achieve and maintain RO system operation at energy-optimal conditions. First, a reverse osmosis water desalination system model is derived from mass and energy balances. Next, the system model is used in conjunction with the system energy usage analysis equations developed in [95, 96] to design an energy optimization-based controller. This controller uses multiple system variables and a user defined permeate production rate to calculate the optimal operating set-points that minimize the specific energy consumption of the reverse osmosis desalination system and satisfy the process and control system constraints. The optimization-based control system is then implemented in a multi-tiered fashion on the UCLA M3 experimental RO system (detailed in Chapter 2).

5.2 System Specific Energy Consumption Model

The operation of the M3 experimental RO desalination system was described by the model equations presented in Chapter 3 (Eqs. 3.13 and 3.14). The system parameters K_m and V were calculated from experimental step-test data (as presented in Chapter 3, Section 3.2) and the resulting parameter values are given in Table 3.1. Other model parameters, which are known properties of the experimental system (ρ, A_p, A_m), are also listed in Table 3.1 (in the case of the M3 system where $A_p^f = A_p^r = A_p^p = A_p$).

The specific energy consumption (or SEC), used to determine the energy usage of the reverse osmosis desalination system, is defined as [95]:

$$SEC = \frac{\Delta P}{Y} = \frac{P_{sys} - P_{raw}}{Y} = \frac{P_{sys} Q_f}{Q_p} \qquad (5.1)$$

where ΔP (in Pa) is the pressure generated by the high-pressure pump (the absolute pressure at the entrance to the RO membranes, P_{sys}, minus the pressure of the raw water, P_{raw}, which is equal to the atmospheric pressure) and Y is the water recovery as defined in Eq. 3.18 (the feed and permeate flow rates Q_f and Q_p, respectively, are given in m^3/s). The SEC can be normalized with respect to the feed osmotic pressure (π_0, given in Pa) as follows:

$$SEC_{norm} = \frac{SEC}{\pi_0} \qquad (5.2)$$

In this model formulation for the evaluation of system energy usage, it is assumed that the salt rejection of the membranes is equal to unity. In addition to the RO system model, the correlation between valve resistance and the valve setting is determined as in Chapter 4 (Eqs. 4.2-4.4); the valve position (O_p, in percent) is related to the valve resistance by five logarithmic relations, each used in a different range of valve resistance values (or equivalently, a different range of valve positions). The logarithmic relation was represented by:

$$e_{vr} = \alpha \ln O_p + \beta \tag{5.3}$$

where α (in 1/% open) and β (dimensionless) are the proportionality constant and offset, respectively, for the relation of valve position to valve resistance. Values for α and β for each valve position range are provided in Table 5.1.

The correlations, along with the manufacturer's suggested valve curve (see Appendix B for additional details regarding the valve), are shown in Fig. 4.1. It should be noted that the valve in these experiments was limited to 70% of its full range due to the fact that the 10-turn valve is actuated with a 7-turn motor. This limitation had no effect on the experimental results since the majority of the valve's effectiveness occurs in the lower range of valve positions ($\sim$0-20%) and all of the valve set-points from the nonlinear optimization were well below the upper valve position limit.

Table 5.1: Logarithmic correlation parameters for conversion of valve position to valve resistance.

Position Range (% Open)	α	β
0% to 0.7%	-512287	-167284
0.7% to 1.4%	-12425	11043
1.4% to 7%	-2052	7434
7% to 49%	-1436	6092
49% to 70%	-265	1554

5.3 Optimization-based Control for Specific Energy Consumption Minimization

The energy-optimal controller was designed in order to facilitate system operation at the point of minimum specific energy consumption (for a given rate of water production). As detailed in Chapter 2, the experimental system uses data acquisition software to record sensor data in real time. This RO system user interface program is linked with an optimization code (implemented in MATLAB) that performs the energy optimization, which then sends the controller/actuator set-points back to the RO system user interface to be implemented on the system. As depicted in Fig. 5.1, the RO system user interface receives measurements of feed conductivity, feed flow

rate, feed pressure, retentate flow rate, and retentate concentration from the process measurement sensors. The optimization code receives the sensor data from the M3 system software and carries out the constrained optimization algorithm (see Eq. 5.10 below). When the optimal values for feed flow rate set-point and valve position are determined, these values are passed back to the M3 system software and implemented on the system. Since a specific permeate production rate is commonly an operating requirement for RO desalination systems, the optimization algorithm is constrained to finding the optimal control values that satisfy the user-defined permeate production rate set-point. This value for permeate flow rate set-point is user-specified on the system user interface and is also transmitted to the optimization algorithm along with the sensor data.

The objective of the optimization algorithm is to determine the values of feed flow rate (v_f, given in m/s) and retentate valve resistance (e_{vr}, dimensionless) such that the SEC at the operating condition is minimized and appropriate constraints are satisfied. Substituting Eq. 3.13 into Eq. 3.14 and solving for P_{sys} (given in Pa), the optimization problem of minimizing the SEC can be represented as:

$$\min_{v_f,e_{vr}} SEC = \min_{v_f,e_{vr}} \frac{\Delta P}{Y} = \min_{v_f,e_{vr}} \frac{\rho e_{vr}(v_f - v_p)^2 v_f}{2v_p} \tag{5.4}$$

Furthermore, during the optimization, several constraints are imposed; the first of which dictates that a constant permeate production rate is ensured:
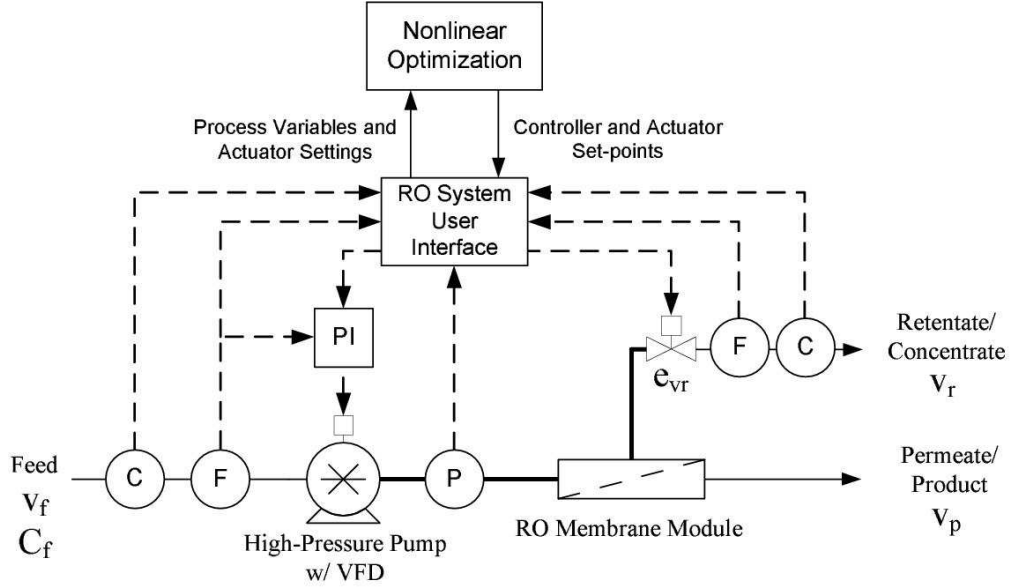
Figure 5.1: Control diagram detailing data flow between measurement sensors, controllers, actuators, RO system user interface and optimization algorithm.

$$v_p = v_p^{set} \tag{5.5}$$

where v_p^{set} (given in m/s) is the overall permeate velocity set-point (v_p is proportional to Q_p through the permeate flow-meter fitting pipe cross-sectional area, A_p). Even though maintaining a specific permeate flow rate will constrain the system, a specified permeate production is often necessary since most RO systems are built to address specific demands for water production. The next constraints (Eqs. 5.6 and 5.7) ensure that the actuator set-points are positive (negative values have no physical meaning in the experimental system, and the output of negative values may damage the actuators). For the feed flow rate constraint, it is assumed that the feed flow rate is

greater than or equal to the permeate flow rate for any reasonable operating condition (no back-flow into the modules through the retentate stream). These constraints dictate that the feed flow rate and the valve resistance (and therefore, the valve position) must be positive:

$$v_f > 0 \tag{5.6}$$

$$e_{vr} > 0 \tag{5.7}$$

It is also necessary to constrain the SEC values to be positive in order to achieve the correct optimization variables. This constraint is represented as:

$$SEC \geq 0 \tag{5.8}$$

Additionally, it is required that the system pressure be greater than the osmotic pressure at the exit of the RO module. If this condition is not satisfied, part of the membrane surface area near the exit region of the module would not be utilized to produce permeate water; the osmotic pressure being above applied pressure. Operation in this region where the transmembrane pressure at the exit region is below the osmotic pressure difference is undesirable, and the process is constrained to operate at or above this limit. This constraint is also called the "thermodynamic restriction" as presented in [95], and has the form:

$$P_{sys} \geq \frac{\pi_0}{(1-Y)} \tag{5.9}$$

In summary, the constrained optimization problem that yields energy-optimal values for the feed flow rate and retentate valve resistance can be formulated as follows:

$$\min_{v_f, e_{vr}} SEC = \min_{v_f, e_{vr}} \frac{\rho e_{vr}(v_f - v_p)^2 v_f}{2 v_p}$$

$$v_p = v_p^{set}$$

$$v_f > 0$$

$$e_{vr} > 0 \tag{5.10}$$

$$SEC \geq 0$$

$$P_{sys} \geq \frac{\pi_0}{(1-Y)}$$

$$0 = \frac{P_{sys}}{\rho} - \frac{1}{2} e_{vr}(v_f - v_p)^2$$

On the experimental system, the optimization algorithm conducts multiple steps at every sampling time in order to obtain the control values (v_f and e_{vr}) that minimize the SEC for the given permeate flow rate. A detailed flowchart of this process is given in Fig. 5.2. First, UDP (User Datagram Protocol) ports on the M3 system computer are opened to allow for data transmission between the programs. The M3 system operation software sends sensor measurements of the feed conductivity, feed flow rate, valve position (converted to valve resistance value as described in Eq. 5.3), and permeate flow rate set-point to the optimization algorithm approximately every

94
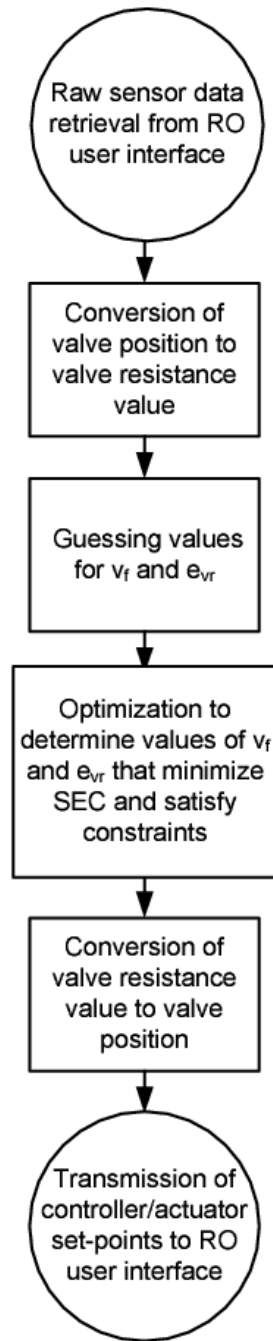
Figure 5.2: Energy optimization decision process conducted at each sampling time.

10 seconds. This timestep is dependent on the time taken to conduct the optimiza-tion; after the first step, each iteration takes between 1-5 seconds (for the optimization conducted in this work). This timestep can be easily tailored to a different RO sys-tem in which the optimization may take longer due to larger disturbances in the feed or faster changes in system set-points; it may also be allowable to conduct the optimization with lower frequency (depending on the requirements of the system). This repeated real-time optimization is particularly useful in systems where the feed concentration is highly variable or in situations where the target permeate produc-tion rate may change over time. After the optimization algorithm receives the raw sensor data and current actuator set-points, the valve position is converted to a valve resistance value (using Eq. 5.3) for use along with the current feed flow rate as initial guesses in the sequential quadratic programming (SQP) optimization algorithm (af-ter the first iteration, the previous optimal values are used in order to provide faster convergence). The system model is then used to calculate the retentate flow rate, v_r, permeate flow rate, v_p, system recovery, Y, system pressure, P_{sys}, and finally the SEC. The calculated variables are checked against the constraints (Eqs. 5.5 - 5.9); if the constraints are not satisfied, the SQP algorithm determines new control values and repeats the process. If the constraints are satisfied, the optimization algorithm determines if the SEC is at its minimum with respect to the feed flow rate and system recovery; if not, the SQP algorithm determines new control values and repeats the process. If the constraints are satisfied and the SEC is then considered minimized,

the resulting valve resistance is converted to a valve position and the optimal control values are transmitted to the M3 system operation software via the UDP port.

Once the optimization algorithm calculates a feed flow rate set-point and transmits it to the M3 system operation software, the PI controller uses measurements of the feed flow rate to adjust the variable frequency drive (VFD) in order to achieve the desired feed flow rate set-point. The controller takes the form:

$$VFD_{set} = K_f(Q_f^{sp} - Q_f) + \frac{K_f}{\tau_f} \int_0^t (Q_f^{sp} - Q_f)d\tau \tag{5.11}$$

where VFD_{set} is the variable frequency drive setting (Volts), Q_f^{sp} is the feed flow rate set-point (GPM), and Q_f is the measured feed flow rate (in GPM). In this work, the proportional gain and integral time constant are equal to $K_f = 0.05\ V/GPM$ and $\tau_f = 0.025\ s$, respectively. Finally, with the VFD PI controller operating, the M3 system operation software applies the actuated valve position retrieved from the optimization code to the actuated retentate valve on the experimental system.

It is noted that using the UDP data transmission is also advantageous because the computer running the optimization algorithm can be remotely located. This arrangement would allow for centralized energy optimization of multiple RO systems that are spatially distributed, or could allow for a multi-objective optimization formulation that takes into account the production and conditions of multiple RO systems.

5.4　Experimental Procedure

The M3 RO system was initially turned on and the PI controller for feed flow rate was activated. When the system is operated, the permeate and retentate streams are recycled back to the feed tank (full-recycle mode). When the system is operated in the full-recycle mode, the feed concentration will not increase over time, since both the permeate and retentate streams are returned back to the feed tank. Such a concentration increase would only happen if the permeate stream was collected in a separate vessel and only the retentate stream was recycled. Feed solutions with several NaCl concentrations were used (1600 ppm, 1850 ppm, and 3500 ppm) for the experiments presented in this work at pressures ranging from 110 to 170 psi. These operating conditions are typical for the feed salinities presented; however, other brackish water systems (5000-35000 ppm TDS) and also seawater desalination systems (typically 35000+ ppm TDS) will operate at higher pressures and usually lower recoveries. After the system reached a steady-state, the nonlinear optimization program was activated to begin transmitting the optimal set-point values to the RO system interface. After the set-point values were received by the RO user interface, the set-points were implemented on the actuated retentate valve and the PI controller on the feed pump. The sensor data taken from the experimental system were averaged (before transmission to the optimization code) using a 19 point moving average to remove the majority of the sensor noise. To obtain the other experimental sub-optimal data points (points

at higher SEC values), the system was manually adjusted to achieve a range of feed pressures and feed flow rates while maintaining the desired constant permeate flow rate. This process was conducted in order to demonstrate the accuracy of the energy usage model at sub-optimal operating conditions (operating points with higher SEC than the optimal operating point, since the optimization only provides the optimal set-point values for the actuator/controller).

After the experimental data (system pressure and flow rates) is collected, the theoretically predicted SEC (at a fixed permeate flow rate) for the range of experimental operating recovery is calculated from the following equation [95]:

$$SEC_{norm} = \frac{\rho Q_p}{A_m K_m \pi_0} + R \left(\frac{ln \left(\frac{1}{1-Y} \right)}{Y^2} \right) \tag{5.12}$$

where the permeate flow rate (Q_p, m^3/s) was measured by the permeate flow meter, the system recovery (Y, dimensionless) was determined through the feed, concentrate, and permeate flow measurements, the feed osmotic pressure (π_0, in Pa) was calculated using measurements of the feed conductivity, and the constants A_m and K_m (active membrane area in m^2 and feed-side mass transfer coefficient in m/s, respectively) were determined through model verification (as described in Chapter 3.2). In the experimental results, the theoretical curves for two cases were presented; in the first case, the membrane rejection (R, dimensionless), was assumed to be unity, and in the second case, R was set equal to the observed rejection calculated from the

experimental data ($R = R_0 = 1 - C_p/C_f$ as defined in Chapter 1.1).

The thermodynamic restriction curve was also plotted along with the experimental data and theoretical operating curves. The equation for the normalized SEC at the thermodynamic restriction (SEC_{norm}^{tr}) is given by [95]:

$$SEC_{norm}^{tr} = \frac{R}{Y(1-Y)} \tag{5.13}$$

Comparing the experimental data to the theoretically predicted specific energy consumption curves provides an indication of how far the experimental data falls from the model predictions (with and without accounting for membrane salt rejection). It is also noted that the thermodynamic restriction curve of Eq. 5.13 does not include a constraint on the permeate flow rate (as is the case in Eq. 5.12), and represents the absolute minimum energy usage possible at a given recovery (for any achievable combinations of feed flow rate, valve resistance, and permeate flow rate) where the thermodynamic restriction is satisfied. Therefore, the primary objective was to compare the experimental operating points to the constrained theoretical SEC (represented in Eq. 5.12) and determine the accuracy of model predictions when operating at a fixed permeate production rate.

5.5   Results and Discussion

When the system is operating at a fixed permeate flow rate, there exists only one degree of freedom with respect to the operating point. If the feed flow rate is changed, the pressure must take on a specific value to ensure that the permeate flow rate remains constant. The converse is true; if the system pressure is changed, the feed flow rate must take on a specific value in order to maintain the desired permeate flow rate. Because of this, for each normalized permeate flow rate, a single curve exists to describe the specific energy consumption at various recovery values [95]. The set of experiments presented in Figs. 5.3 - 5.7 demonstrates the experimental system's performance compared to the system performance as predicted by the model for various feed solution salt concentrations. Results for two of the feed solutions (1600 ppm and 1850 ppm) are given for two different permeate flow rate set-points (1 gpm and 1.45 gpm). In the presentation of the results, the SEC is normalized to SEC_{norm} as presented in Eq. 5.2.

From Figs. 5.3 - 5.7, it can be seen that the experimental system operating points are very close to the theoretically predicted operating points (in terms of specific energy consumption and recovery), for the ideal case of 100% salt rejection by the membranes. It can also be seen that no experimental operating points are shown for higher recoveries than the energy optimal operating point dictated by the controller in order to demonstrate the existence of the minimum specific energy consumption.
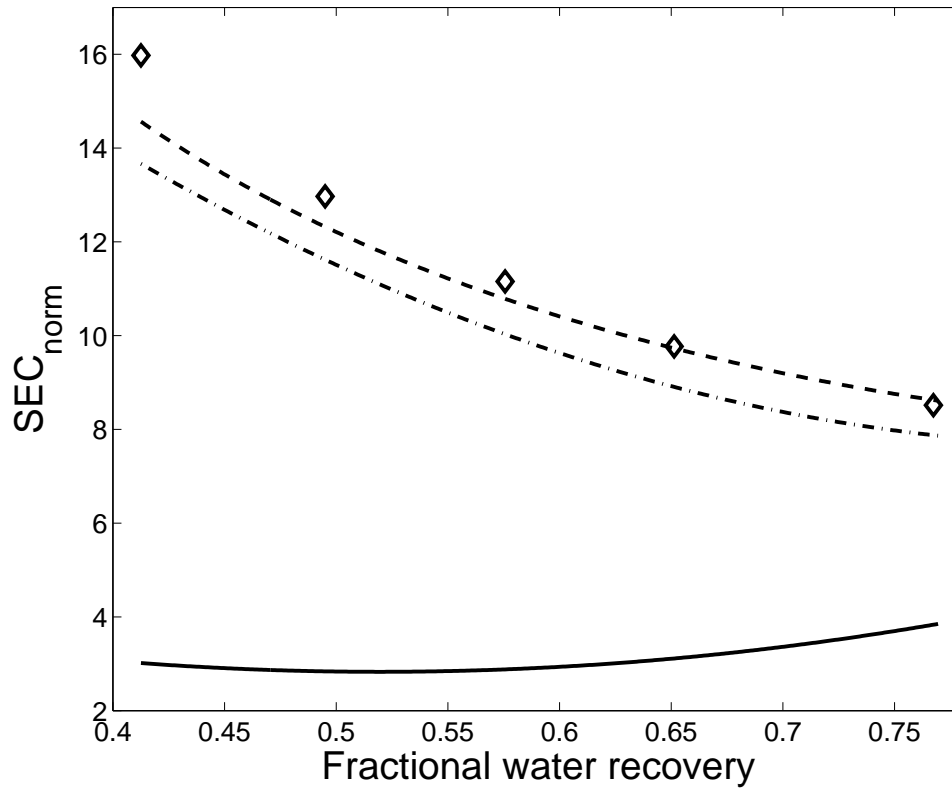
Figure 5.3: RO system normalized specific energy consumption with respect to fractional water recovery at a fixed permeate flow rate of 1 gpm and a feed salt concentration of 1600 ppm; the dashed line represents the theoretical operating curve assuming 100% salt rejection by the membranes (determined through Eq. 5.12), the dash-dotted line represents the theoretical operating curve accounting for membrane salt rejection (determined through Eq. 5.12), the diamonds represent experimental system data, and the solid line represents the thermodynamic restriction for complete salt rejection (determined through Eq. 5.13). Feed pressure to the RO modules ranged from 112 to 116 psi.
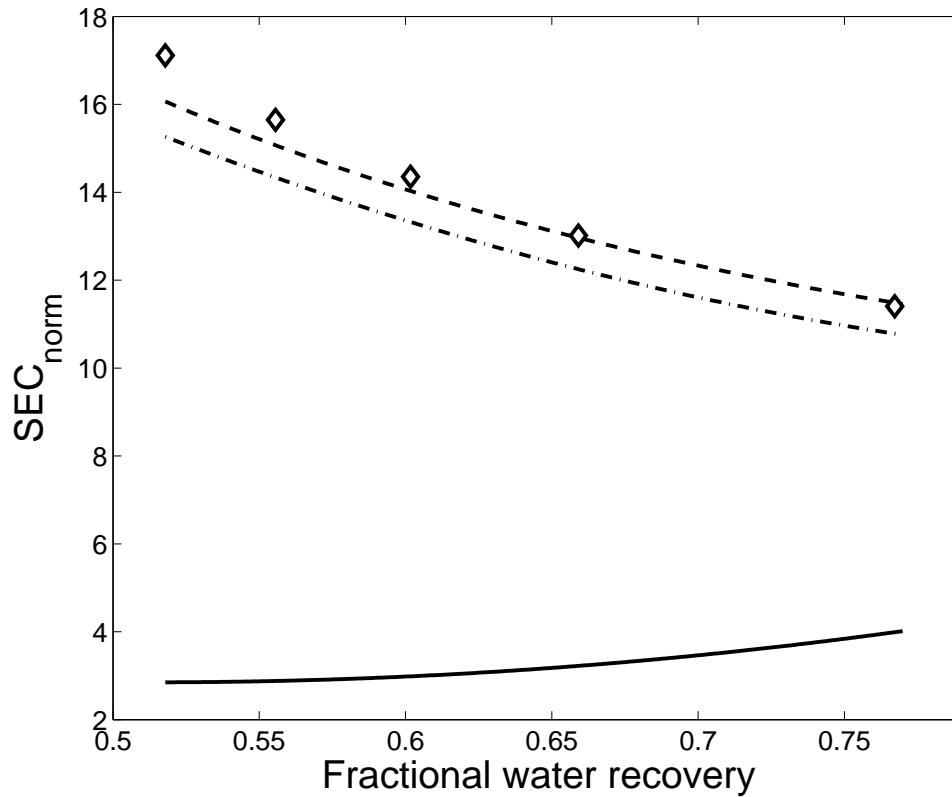
Figure 5.4: RO system normalized specific energy consumption with respect to fractional water recovery at a fixed permeate flow rate of 1.45 gpm and a feed salt concentration of 1600 ppm; the dashed line represents the theoretical operating curve assuming 100% salt rejection by the membranes (determined through Eq. 5.12), the dash-dotted line represents the theoretical operating curve accounting for membrane salt rejection (determined through Eq. 5.12), the diamonds represent experimental system data, and the solid line represents the thermodynamic restriction for complete salt rejection (determined through Eq. 5.13). Feed pressure to the RO modules ranged from 151 to 156 psi.
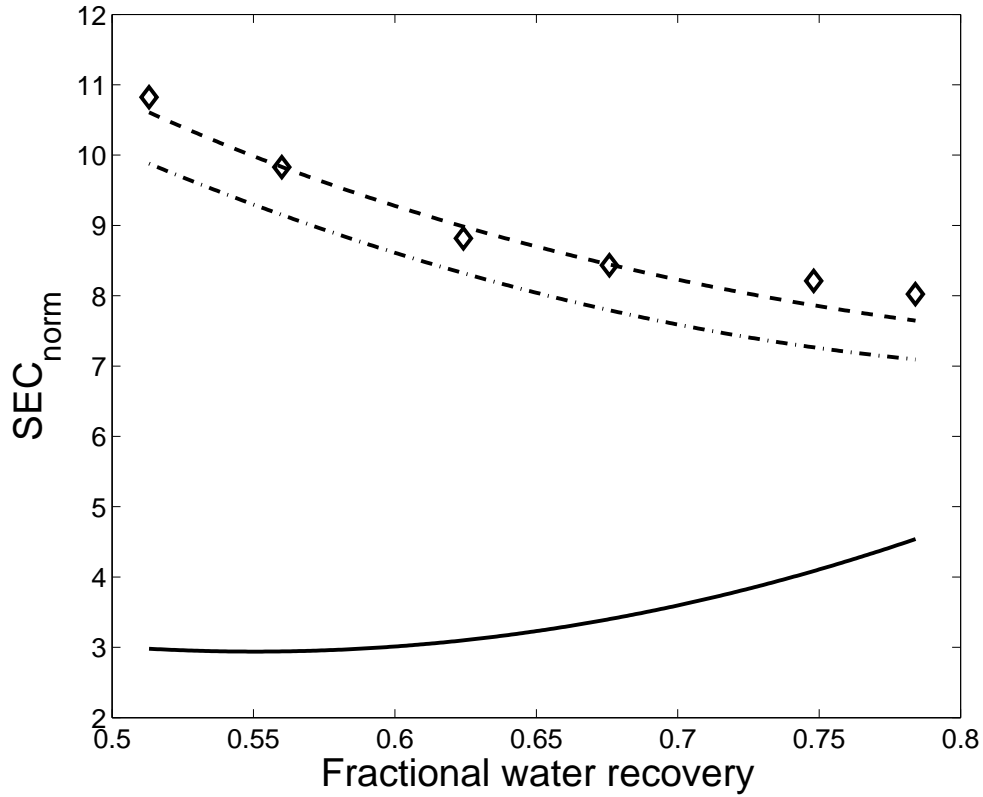
Figure 5.5: RO system normalized specific energy consumption with respect to fractional water recovery at a fixed permeate flow rate of 1 gpm and a feed salt concentration of 1850 ppm; the dashed line represents the theoretical operating curve assuming 100% salt rejection by the membranes (determined through Eq. 5.12), the dash-dotted line represents the theoretical operating curve accounting for membrane salt rejection (determined through Eq. 5.12), the diamonds represent experimental system data, and the solid line represents the thermodynamic restriction for complete salt rejection (determined through Eq. 5.13). Feed pressure to the RO modules ranged from 112 to 128 psi.
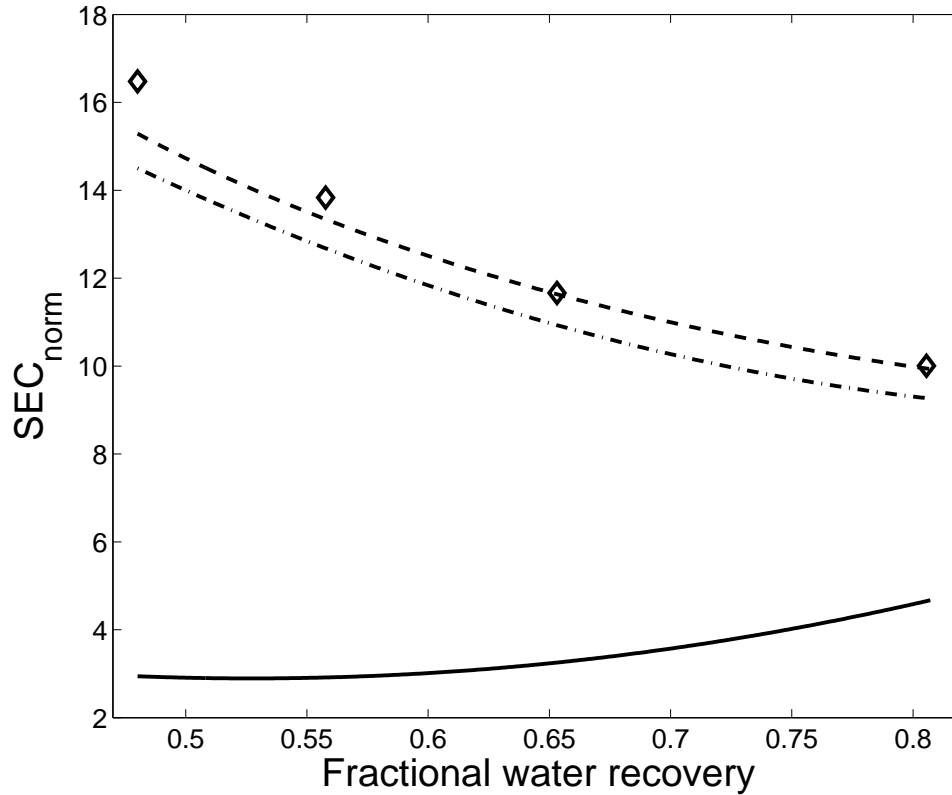
Figure 5.6: RO system normalized specific energy consumption with respect to fractional water recovery at a fixed permeate flow rate of 1.45 gpm and a feed salt concentration of 1850 ppm; the dashed line represents the theoretical operating curve assuming 100% salt rejection by the membranes (determined through Eq. 5.12), the dash-dotted line represents the theoretical operating curve accounting for membrane salt rejection (determined through Eq. 5.12), the diamonds represent experimental system data, and the solid line represents the thermodynamic restriction for complete salt rejection (determined through Eq. 5.13). Feed pressure to the RO modules ranged from 155 to 164 psi.
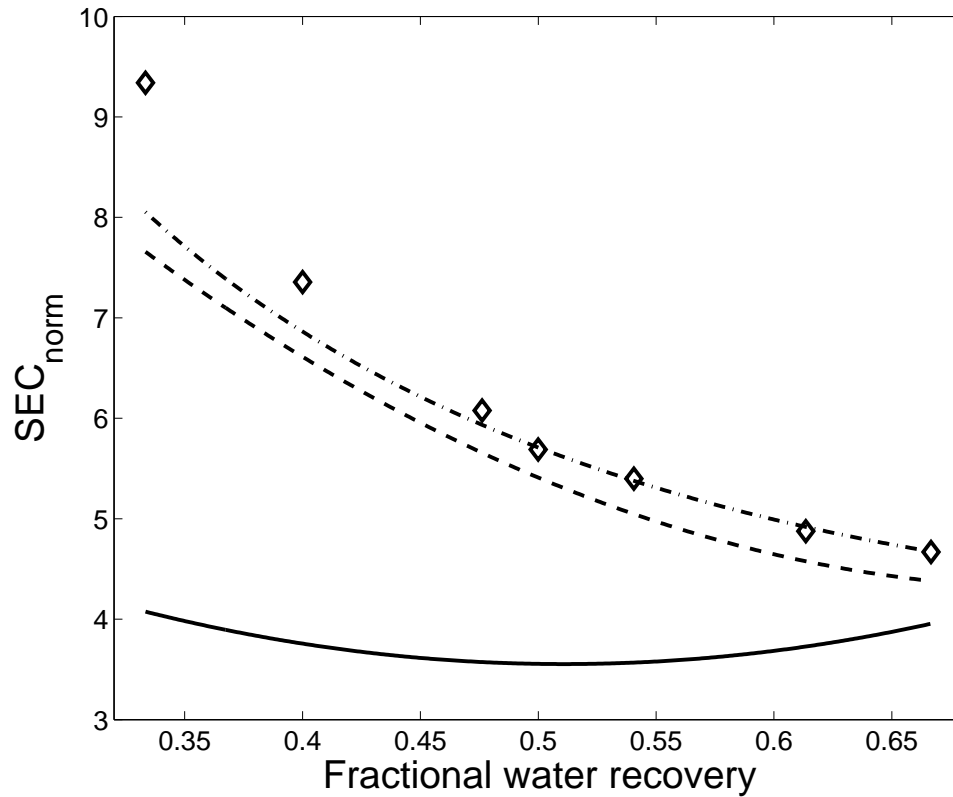
Figure 5.7: RO system normalized specific energy consumption with respect to fractional water recovery at a fixed permeate flow rate of 1 gpm and a feed salt concentration of 3500 ppm; the dashed line represents the theoretical operating curve assuming 100% salt rejection by the membranes (determined through Eq. 5.12), the dash-dotted line represents the theoretical operating curve accounting for membrane salt rejection (determined through Eq. 5.12), the diamonds represent experimental system data, and the solid line represents the thermodynamic restriction for complete salt rejection (determined through Eq. 5.13). Feed pressure to the RO modules ranged from 111 to 136 psi.

This is due to the fact that the minimum of the theoretical SEC curve occurs at a point where the physical limitations of the system (valve settings below 1% result in unpredictable valve performance and rapid increase in system pressure) components prevent experimental system operation at higher recoveries than the optimal one while maintaining the desired permeate flow rate set-point. In Fig. 5.7, an increasing deviation between experimental results and theoretical predictions at decreasing recovery values can be observed (observed errors of 9-20% between experimental data and theoretical prediction accounting for observed salt rejection at lowest recovery values and observed errors of 1-13% at the highest attempted recovery values). This is due to the fact that when examining the equation for SEC (Eq. 5.1), it can be seen that at decreasing system recoveries, experimental errors (sensor noise, etc.) on the recovery value have an increasing effect on the calculated SEC value since the recovery appears in the denominator. For example, if the actual recovery is 0.15 and the measured recovery is 0.1, the error in the SEC value would be 33%; in the case where the actual recovery is 0.85 and the measured recovery is 0.8 (same absolute deviation of 0.05), the error in the SEC value would be only 6%.

Another issue with the comparison of the theoretical and experimental minimum SEC points is that the resulting permeate salt concentration generally increases with system recovery and can rise above the maximum allowable permeate salt concentration (in this work, it is desired that the water produced maintains a salt concentration below 500 mg/L (ppm)). In Fig. 5.8, it can be seen that the permeate concentration

remains under or near the limit of 500 ppm for the feed solutions with lower salt concentration; however, for the feed solution containing 3500 ppm of NaCl, the permeate salt concentration was above 500 ppm for the highest experimental recovery point. This issue arose because the salt rejection (fraction of salt retained on the feed side of the membrane channel) of the membranes was not constant as assumed in the optimization problem in Eq. 5.10. It was also found that the membrane salt rejection was not constant at a given water recovery for different feed solution conditions. The salt concentration in the permeate stream increases with increasing recovery, which is observed in the data in Fig. 5.8.

Since the rejection is a complex function of the feed flow rate and of the transmembrane pressure, it is very difficult to include this expression in the model as a constraint since its explicit functional form is not available for any experimental RO system. The rejection is also dependent on (but not limited to) membrane structure, membrane composition, temperature, and the type of ions present in solution. Through these considerations, it can be seen that the rejection will be a complex function of the feed flow rate and of the transmembrane pressure that is unique to each reverse osmosis desalination system [38]. If the rejection can be explicitly described in terms of feed flow rate and transmembrane pressure, then this expression can be used to theoretically determine the permeate concentration at any operating point. An explicit expression for rejection could also be used in the controller formulation as a constraint; in this way, the optimization code could determine the lowest SEC op-

erating point that satisfies the permeate salt concentration standard. In the present work, the membrane rejections for the different water recoveries corresponding to the operating points were determined experimentally and used to re-compute the theoretical operating curve at the user-specified permeate flow rate. The resulting operating curve, accounting for membrane salt rejection (found to be between 70-75%), is shown in Figs. 5.3 - 5.7 as the dashed line, and is very close to the experimentally computed operating points.

Remark 2: In the most likely situation (as in the current work), the expression for salt rejection dependence on process parameters is not known a priori for a specific RO desalination system; therefore, the general form of the controller can not use the rejection (and subsequently, the permeate concentration) as an explicit constraint in the model-based optimization. It is noted, however, that it is possible to develop specific system data with respect to rejection as a function of operating conditions, as shown in previous work with the M3 RO desalination system (see [38]). Such system specific information can be used to generate suitable empirical correlations (e.g., for rejection) that can serve to improve the online energy optimization algorithm. For the experiments presented in this chapter, the simplifications resulting from the lack of knowledge of the salt rejection expression result in the plant-model mismatch observed in the presented data. This mismatch results in the values of the theoretical prediction for the model with complete salt rejection to be, at times, closer to the experimental data collected. Under these circumstances, additional controller constraints must
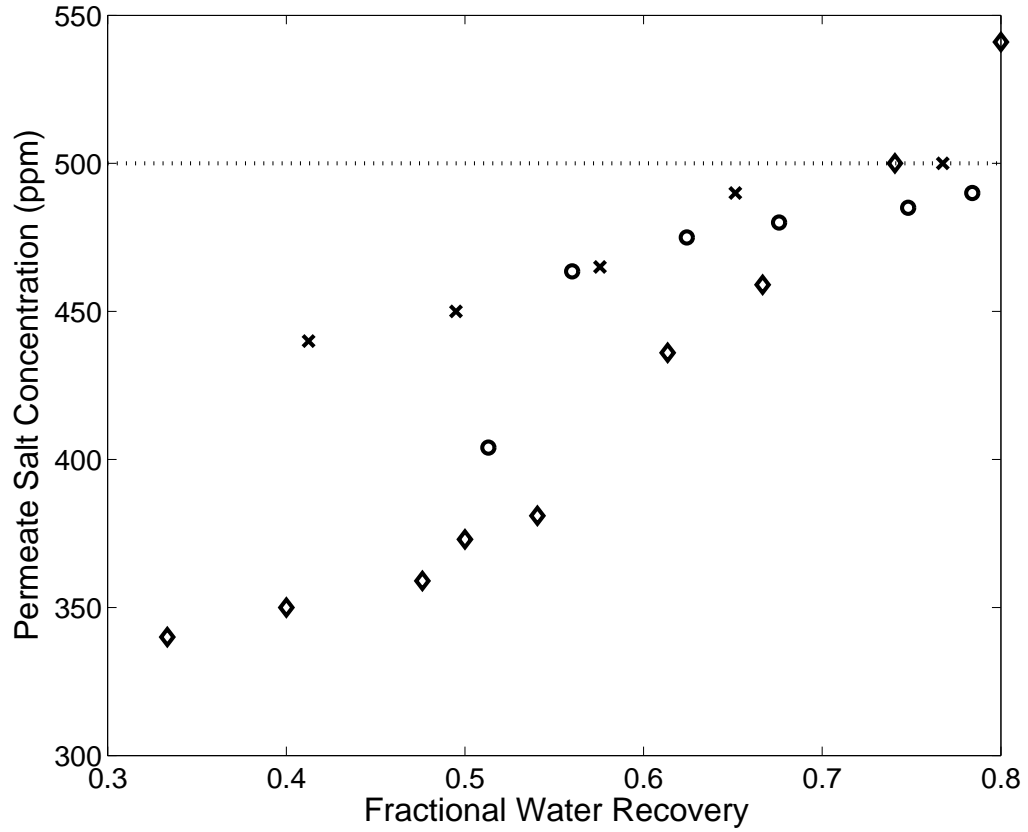
Figure 5.8: Permeate salt concentration with respect to fractional water recovery at a fixed permeate flow rate of 1 gpm; experimental system data from a feed concentration of 3500 ppm (◇), a feed concentration of 1850 ppm (○), and a feed concentration of 1600 ppm (×). The dotted line represents the permeate salt concentration limit of 500 ppm.

be defined in order to "step back" the system operating conditions from the water recovery that gives the optimal SEC to a lower recovery value that provides permeate quality meeting the drinking water standards. It is proposed (for future work) that the system can institute a procedure where, first, the variable frequency drive speed is increased (this will increase the permeate flow rate, ensuring that the total permeate production stays at or above the required value); then, the controller will open the retentate valve until the permeate flow rate drops back to the set-point value. In this way, the recovery should be decreased, while the salt rejection should increase, leading to a lower concentration of salt in the permeate stream. A feedback control loop measuring the permeate salt concentration can be used to determine when the permeate quality has reached the desired level; at this point, the "stepping back" procedure can be stopped.

5.6   Conclusions

An optimization-based control strategy was developed and experimentally implemented on a reverse osmosis (RO) membrane desalination system. First, a non-linear model of the system was derived from first principles and was combined with a model for RO system specific energy consumption to form the basis for the design of an energy-optimization based control system. The model parameters were computed using experimental system step-test data. The model in this chapter assumed that the rejection of the RO membranes was equal to unity, but it is noted that the optimiza-

tion problem can be augmented to take rejection into account. The control system uses real-time sensor data and user-defined permeate flow requirements to compute in real-time the energy-optimal set-points for the retentate valve position and feed flow rate. Implementation of the control system on UCLA's experimental M3 RO system demonstrated the ability to achieve energy-optimal operation that nearly matched the theoretically predicted energy consumption curves, with a maximum deviation of 20% or less at the lower recovery levels ($\leq 45\%$).

Chapter 6

RO Membrane Monitoring and Image Analysis for Scale Control

6.1 Overview

In the previous chapters, several types of overall system control (control of system pressure, flow rates, system recovery) were developed in order to minimize the energy consumption of RO systems, provide methods for conducting efficient set-point changes, and maintain system operation at a user-specified operating point in the presence of feed water quality disturbances. However, in the case of high-recovery system operation, these control methods do not explicitly account for the effects of mineral salt scaling on the RO membrane surface (this must also be considered as a process constraint in RO system operation). This chapter discusses the issues associated with mineral salt scaling on the RO membrane surface (primarily in a brackish water setting), a novel method of visual membrane mineral scale detection, and the development and implementation of an image analysis software for real-time analysis

of the images provided from the membrane scale monitor. After the image analysis software determines the level of mineral salt scaling/fouling on the RO membrane surface, the results are used in an automated fashion to implement a method of scale mitigation called feed flow reversal (FFR). It is then shown, for the first time, that this novel method of membrane monitoring (coupled with the real-time image analysis software) is able to actuate FFR and mitigate mineral salt scaling on the RO membrane surface over an extended period of time.

6.2   Introduction

In recent years, reverse osmosis (RO) desalination has emerged as a leading method for desalting seawater, inland brackish water and water for water reuse applications ([73], [46], [37]). In inland desalination of brackish water and water reuse, concentrate (brine) management is a major challenge given the limited options for concentrate disposal. With increasing product water recovery, the volume of the residual concentrate stream is reduced, increasing the available options for management of this stream (i.e., treatment and disposal).

Because the costs associated with managing residual desalination concentrate are typically high (especially at inland locations), high levels of product water recovery (85-95%) are often required for optimal inland desalting operation ([32], [95]). As the permeate recovery level increases, the level of concentration polarization (i.e.,

increased solute concentration at the membrane surface relative to the bulk) rises, increasing the propensity for membrane fouling and scaling [94]. Mineral scaling can occur when the concentrations of sparingly soluble dissolved mineral salts (e.g., gypsum ($CaSO_4 \cdot 2H_2O$), $BaSO_4$, $SrSO_4$, $CaCO_3$, SiO_2, etc.) near membrane surfaces rise above their solubility limits. As a consequence, sparingly soluble mineral salts can precipitate in the bulk and subsequently deposit onto the membrane surface as well as crystallize directly on the membrane. Mineral scaling can lead to a significant reduction in membrane performance (e.g., flux reduction and salt rejection impairment) and shortening of membrane life, thereby increasing process cost and imposing operational limits on the achievable product water recoveries [37].

The most common feed water conditioning methods for mitigating mineral scale formation are feed water pH adjustment (primarily for carbonate minerals) and antiscalant treatment [73]. Antiscalant treatment involves dosing of antiscalant chemicals that kinetically delay the onset of mineral salt crystallization and may also retard the growth of mineral salt crystals [67]. The use of antiscalant requires precise knowledge of the optimal antiscalant dose to provide adequate scale protection, decrease process cost associated with antiscalant use, and ensure that excessive antiscalant dosages do not lead to increased scaling or biofouling [75].

If scale formation is detected at an early stage, membrane cleaning can be effectively accomplished via chemical cleaning [13], osmotic backwash [15] or feed flow reversal [17]. Also, conservative operation (i.e., low recovery) can be imposed to

ensure that mineral salt concentrations at the membrane surface are below saturation. The above approaches require knowledge of the scaling propensity for the source water and at the operating conditions (e.g., based on feed water chemistry data or scale monitoring) or real-time membrane monitoring information to detect the onset of mineral scaling. With feed water that may fluctuate in its composition, changes in the level of supersaturation of sparingly soluble mineral salts will also vary with time. Unfortunately, present water quality monitoring capabilities do not provide adequate real-time information on the concentration of mineral salt scale precursors. Therefore, it is imperative to establish when mineral scale may occur and/or directly detect the onset of mineral scaling in order to determine the appropriate frequency of needed membrane cleaning. Early mineral scale detection is also instrumental in determining required adjustments in operating conditions (e.g., recovery level and antiscalant dose) to ensure process operation below the scaling threshold.

Methods of RO system control that can respond to changes in feed water salinity and composition have been proposed ([16], [14], [3]; and references therein). However, even with such systems there is a need for early detection and monitoring of membrane mineral scaling. In-situ monitoring techniques based on indirect ultrasonic crystal detection have been evaluated for the study of mineral scaling and for use in mineral scale and fouling detection in RO systems ([94], [26], [87], [50]). These methods respond to the buildup of a foulant layer but do not differentiate the type of fouling layer (e.g., particulate deposition, bacteria or surface formed crystals) and in

general do not detect with sufficient reliability the early onset of membrane surface crystallization (i.e., first formed mineral crystals) ([86], [85], [59]). Extensive research has also been conducted on modeling flow patterns (through CFD) and concentration polarization in different types of RO membrane processes [57, 51, 12, 11, 81, 7].

In order to accomplish the goal of early mineral scale detection on the membrane surface, a novel ex-situ membrane monitor (MeMo) that enables real-time membrane surface imaging has been recently developed ([58], [84]). The MeMo detector enables direct observation (i.e., real-time images) of a representative membrane surface where the operating conditions in this ex-situ RO cell are matched to the conditions in the selected spiral-wound element (e.g., tail or lead elements) of the RO plant.

In this work, the application of online imaging of mineral scale formation using the MeMo type system is demonstrated making use of specialized novel image analysis software for real-time measurement of the fractional area of the membrane surface that is covered by mineral scaling, as well as the number of crystals present on the membrane surface. Using information from real-time mineral scale measurements, it is then possible to automate control actions (e.g., initiation of cleaning strategy, adjustment of antiscalant dose or adjustment of product water recovery) based on a user-defined mineral scaling threshold.

6.3 Experimental

6.3.1 Materials

Salt solutions were prepared using calcium chloride ($CaCl_2 \cdot 2H_2O$), barium chloride

($BaCl_2 \cdot 2H_2O$), magnesium sulfate ($MgSO_4 \cdot 7H_2O$), and anhydrous sodium sul-

fate (Na_2SO_4), all reagent grade obtained from Fisher Scientific (Pittsburgh, PA).

The solutions were prepared in de-ionized water obtained by filtering distilled water

through a Milli-Q Water System (Millipore Corp., San Jose, CA).

Table 6.1: Solution composition for membrane scaling experiments.

Ion	Concentration
	(mg/L)
Na^+	636
Ca^{2+}	681
Mg^{2+}	276
Ba^{2+}	1.2
Cl^-	1206
SO_4^{2+}	2419

The solution composition for the present study (Table 6.1) mimicked water com-

position of primary RO concentrate (salinity of 5219 mg/L total dissolved solids)

118

that would result from desalination of Colorado River water at recovery of 85% [47]. The degree of supersaturation of the various solutions with respect to gypsum ($CaSO_4 \cdot 2H_2O$) and barite ($BaSO_4$) was quantified in terms of the respective saturation indices for these salts (i.e., SI_g and SI_b,):

$$SI_g = \frac{(Ca^{2+}) \cdot (SO_4^{2-})}{K_{sp,g}} \qquad (6.1)$$

$$SI_b = \frac{(Ba^{2+}) \cdot (SO_4^{2-})}{K_{sp,b}} \qquad (6.2)$$

where (Ca^{2+}), (Ba^{2+}) and (SO_4^{2-}) are the activities of the calcium, barium and sulfate ions, respectively, and $K_{sp,g}$ and $K_{sp,b}$ are the solubility products for gypsum and barite, respectively. The saturation indices were determined via multi-electrolyte thermodynamic solubility calculations using the OLI Analyzer software [1]. The saturation indices of gypsum and barium sulfate (at $25°C$) for the feed solution (Table 6.1) were determined to be 0.99 and 156, respectively at the solution pH of 7.

The two commercial antiscalants were used to demonstrate detection capability under the action of scale retardation were PC-504 (Nalco, Naperville, IL) and Flocon 260 (Biolab Water Additives, Tucker, GA), hereinafter referred to as AS1 and AS2, respectively. Ethylenediamine-tetraacetic acid (EDTA, Fisher Scientific, Pittsburgh, PA) was the cleaning agent for the system components. All scaling tests were performed using the TFC-ULP membrane (Koch Membrane Systems, San Diego, CA),

119

reported to have a root mean-square surface roughness of 54.2 nm, $L_p \times 10^7 = 12.3$

0.7 (m $bar^{-1}s^{-1}$) and nominal salt rejection of 97% [47].

6.3.2 Mineral Scale Detection System

The membrane monitor (MeMo) used in the present work is similar in construction
to the ex-situ scale observation detector (EXSOD) previously developed at UCLA
([28], [84]). Briefly, the MeMo system consists of a semi-transparent plate-and-frame
reverse osmosis cell that allows for real-time imaging of the surface of a membrane
coupon placed in the cell. The MeMo cell is comprised of an opaque base with a
fritted bottom metal plate to allow the permeate water to channel into the permeate
collection tube. The membrane coupon is placed on top of the fritted metal plate;
then, several layers of teflon seals are placed around the membrane coupon to allow
for high-pressure leak-free operation, as well as to create the feed channel (3.16 cm
wide, 8.24 cm long and 2.66 mm in height). A transparent acrylic spacer is placed in
between the teflon seals to allow for the entry of the incident light from a direction
parallel to the membrane's surface. The acrylic spacer and teflon seals are secured
between the base and a thick transparent acrylic block (the top of the cell) which
allows for the real-time observation of the membrane surface during system operation.
The arrangement of the lighting and feed/retentate streams is shown schematically
in Fig. 6.1. In order to provide proper lighting to enable crystal detection, a line
lighting source is fastened to the side of the MeMo cell. Above the cell, a CCD

(charge-coupled device) camera attached to a monoscope focuses on a small portion of the membrane coupon surface.

When used as a monitor for a commercial RO plant, feed water to the MeMo cell would be drawn from the high pressure feed of the tail element of the plant. However, in the present study, a model feed solution (Table 6.1) was delivered to the MeMo cell from a stirred and temperature controlled (within $0.5°C$) feed reservoir (18 L), using a high pressure pump. The feed flow rate is controlled by a pump equipped with a variable frequency drive and a bypass valve before the RO cell. Pressure in the cell is adjusted using an actuated valve on the retentate line (located after the MeMo cell). In the present work, the system was operated in a total recycle mode (i.e., permeate and retentate streams continuously recycled to the feed reservoir). As a precautionary measure, a cartridge filter (0.2 μm nominal pore size) was installed on the retentate side of the membrane modules to trap crystals that may have formed in the membrane channel (Fig. 6.2). Feed flow rate, permeate flux, feed properties (i.e., temperature, pH and salinity) and permeate conductivity were recorded digitally using a computerized data acquisition system as described in [47].

6.3.3 Scaling Experiments

Prior to each scaling test, a new membrane coupon was conditioned for a period of 4 hours by circulating a feed solution composed of all salts except $CaCl_2 \cdot 2H_2O$ and $BaCl_2 \cdot 2H_2O$ through the feed channel with the permeate flow rate set at about 1.2
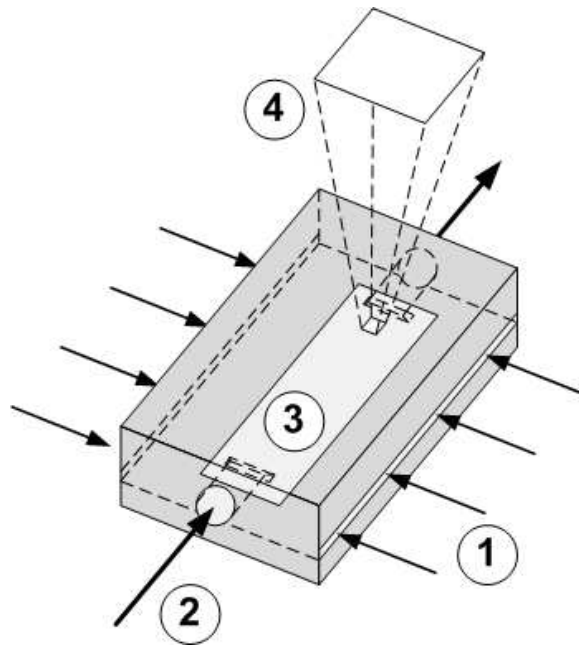
Figure 6.1: Membrane monitor (MeMo) cell: 1) Incident light for crystal detection is parallel to the membrane surface, 2) Feed water stream entry, 3) Membrane coupon underneath transparent top, and 4) Example of imaged portion of the membrane surface (size and position can be adjusted).
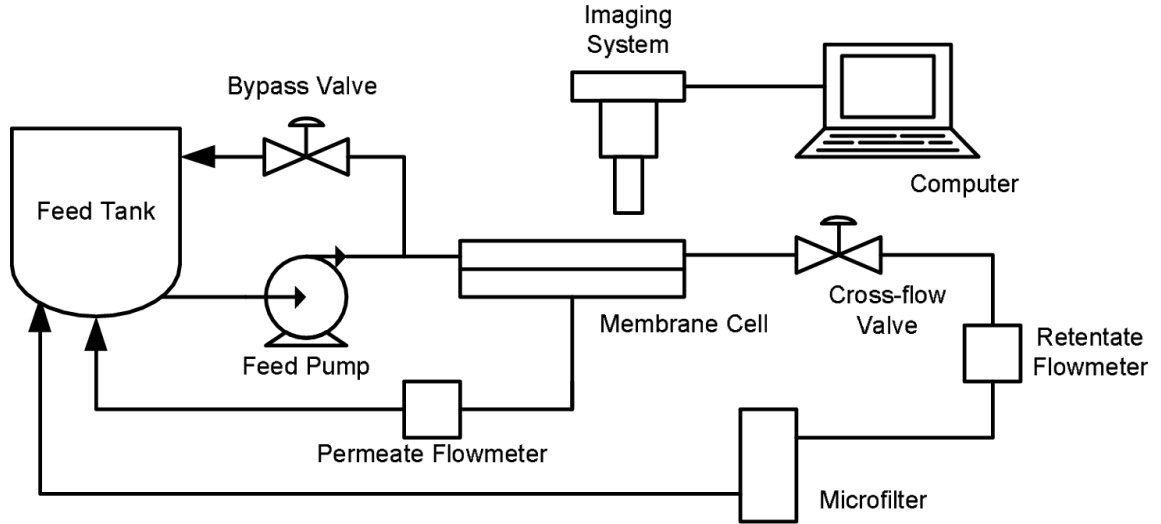
Figure 6.2: Schematic of the membrane monitor (MeMo) testing arrangement.

mL/min (3.64 cm^3/cm^2h). Subsequently, predetermined volumes of stock calcium

chloride and barium chloride solutions were added to the feed reservoir to obtain the

desired feed solution composition (Table 6.1). All scaling experiments were carried

out at 25°C in a total recycle mode at a cross flow velocity of 4.3 cm/s and an initial

permeate flow rate of 1.1 mL/min (or flux of 3.34 cm^3/cm^2h), with transmembrane

pressure typically in the range of $1.03 \times 10^3 - 1.13 \times 10^3$ kPa. At the above conditions,

the average initial solution supersaturation indices (SI) at the membrane surface

were 2.15 and 403 for gypsum and barite, respectively [47]. It is noted that the SI

value for gypsum at the membrane surface was as high as about 2.6 at the channel

exit and about 2.2 in the MeMo's imaged region. The above SI values are average

values based on the concentration profile determined previously using a 3-dimensional

numerical concentration polarization model [57] for the present channel geometry [86]

and operating conditions.

Images were collected from four separate experiments with the MeMo system operating at the conditions described above. The feed salt concentrations were identical for all scaling tests which were conducted with and without antiscalant addition to the feed solution (see Table 6.2).

Table 6.2: List of scaling tests.

Run	Antiscalant	Concentration (ppm)
1	None	0
2	AS1 (PC-504)	1.5
3	AS1 (PC-504)	3
4	AS2 (Flocon 260)	3

6.4   Image Analysis

Membrane surface images were captured and stored on the MeMo data acquisition computer at prescribed time instants. During mineral scaling experiments, the MeMo imaging system was set to automatically capture and store an image of the membrane surface every 15 minutes. Upon capture each image was analyzed using imaging processing software developed specifically for the MeMo system. The membrane surface image analysis (MSIA) software consists of the following components; image pre-

processing/algorithm initialization, image subtraction and smoothing, edge detection and hysteresis thresholding [48], crystal confirmation and crystal count/area calculation, and data output (Fig. 6.3). A description of the image analysis approach is provided in the following sections.

6.4.1 Image Pre-processing/Initialization

The first process in the on-line image analysis is that of image pre-processing and initialization. In the pre-processing step, the captured color image is converted to an unsigned 8-bit grayscale image. The initialization step involves the creation of a "cumulative scaling image" in order to track the overall growth of the crystals (surface area covered and number of crystals) throughout the course of the experiment. The "cumulative scaling image" is a binary image that is of the same pixel dimensions as the images to be analyzed which is used as a buffer to record confirmed instances of mineral scaling on the membrane surface. Once the presence of a crystal on the membrane surface is confirmed, the corresponding pixels on the cumulative scaling image are changed from 0 to 1. This approach allows for tracking of previously detected crystals and crystals that appear to have stopped growing. The cumulative image is then used to determine the fractional scale coverage and crystal count during the crystal confirmation, counting, and area calculation step at the end of each analysis iteration.

6.4.2 Image Subtraction and Smoothing

The purpose of the image subtraction step is to determine where crystals have begun to form or have continued growing. This is accomplished by examining a previous image of the membrane and comparing it to a more recent image. Through subtraction of a reference image from the new image, it is possible to determine which pixels have changed and to what degree; this provides information as to where crystals have formed on the membrane surface. It is noted that subtle changes in lighting or the membrane surface pattern can affect the detection greatly from image to image. The smoothing step is able to decrease the effects of this unwanted "noise" that may lead to false detection of crystals.

In the subtraction step, the new membrane surface image and the reference image (i.e., the membrane surface image preceding the new image) are pre-processed and the absolute difference (necessary to eliminate any bias as to the type of pixel change; darker or lighter) between the images is taken. In this way, the pixels on the images which have changed to the greatest degree become apparent. Image processing filters are then implemented to reduce the impact of possible lighting changes over the course of the experiment. In the smoothing step, the absolute difference image from the subtraction step is processed using several different methods. First, the analysis applies a Gaussian low-pass filter with properties depending on the crystal type and size in the experiment (determined from preliminary experiments with similar feed

water). To smooth the absolute difference image, a convolution of the difference image and of the 2-D Gaussian filter matrix is performed (similar to the procedure of the Canny edge detection method [24]). This Gaussian filter matrix can be represented as:

$$G(x, y) = \frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{6.3}$$

where x and y are the distances from the center of the filter matrix in the horizontal and vertical directions, respectively (the matrix used in this work is a 7×7 matrix, so $x, y \mid x, y \in \mathbb{Z}, |x|, |y| \leq 3$), and $\sigma = 1$.

6.4.3 Edge Detection and Hysteresis Thresholding

To determine where crystals have formed on the surface of the membrane, an edge detection algorithm is used to find the outlines of new crystals or to find the areas of growth for existing crystals. After the edge detection algorithm is applied, hysteresis thresholding is carried out to determine which pixels have changed intensity to a sufficient degree to be considered indicative of crystal formation. After converting the original camera image to grayscale, the pixel values are between 0 (black) and 1 (white). During edge detection and hysteresis thresholding, the pixels take on values relative to the largest intensity change that exists in the image (the pixel which has changed the most from the previous image will have a value of 1, and pixels

that have not changed have a value of 0). In the next step, the smoothed image is then passed through a function that determines the directional gradients of pixel intensity in the processed image (both in the row and column directions). To find the areas of the image where the directional gradient of the pixel values is greatest, each of the directional gradient matrices are squared (element by element) and added together to form an image highlighting the most prominent edges found on the original image. From the resulting image, the pixels where the greatest change in intensity has occurred and the crystal edges can be observed.

Subsequently, hysteresis thresholding is conducted, making use of preset upper and lower thresholds. The upper threshold, denoted as "UT", is used to locate the most prominent changes in pixel value in the processed image; any pixels which have a value larger than the upper threshold are flagged as possible scale pixels. The algorithm then scans any pixels directly connected to the flagged pixels and also flags any of these pixels that have values above the lower threshold (denoted as "LT"). After the corresponding pixels on the binary matrix are marked, the image undergoes several morphological cleaning and filling operations in order to remove isolated pixels (e.g., single flagged pixels surrounded by 8 un-flagged pixels, tending to be a false positive) and also to fill in "holes" (e.g., un-flagged pixels surrounded by 8 flagged pixels).

6.4.4 Crystal Confirmation, Counting, and Area Determination

Before the surface coverage and number of crystals are calculated, the flagged pixels are checked against previously flagged pixels on the cumulative scaling image in order to determine whether or not the newly detected pixels are actually an instance of scaling, or a false positive detection. If a flagged pixel is located in the same spot as in the previous image comparison, then this pixel can be considered to be confirmed scaling. It is also noted that a "confirmation threshold" can be set to allow the user to control how stringent the program will be in confirming the location and amount of scale formed. If the confirmation threshold is preset to a value of 1, this means that a flagged pixel must persist in two consecutive image analyses in order to be considered as scale formation and added to the cumulative scaling image. In systems where the lighting is stable and provides excellent contrast between the membrane and the crystal, this confirmation threshold can be set to zero. For the results presented in this chapter, the confirmation threshold was set to a value of 2.

Since the cumulative scaling image contains only ones and zeros, the flagged pixels are summed and divided by the total number of pixels in the image, resulting in the fraction of the image covered by flagged scaling pixels. This value is then passed to the graphical user interface (GUI) and plotted so that the user can see the real-time update of the surface coverage vs. time (or vs. number of captured images). The

cumulative scaling image is also used in the algorithm to determine the number of crystals present on the membrane surface (this metric can also be used to trigger membrane cleaning). In the crystal count algorithm, the flagged pixel groups are screened using several methods; the first one being an area thresholding process. In this process, crystals with an area smaller than a threshold value are not counted as crystals. Moreover, initially identified potential crystals that did not grow to sizes above the minimum threshold over the course of the experiment were removed from the crystal count and scaled area calculation. For experiments resulting in a smaller number of large crystals (runs 1 and 2), a typical threshold of 350 pixels was set, corresponding to a surface coverage value of about 0.02%. For scaling experiments resulting in a larger number of small crystals, the threshold value was set lower (i.e., 175 pixels for run 3, and 150 pixels for run 4). Subsequent to the area thresholding, a grouping algorithm is applied to the cumulative scaling image. In this procedure, pixels marked as crystals within a pre-determined proximity of other marked pixels are grouped together and considered as part of the same crystal. This pixel proximity distance is based on the average radius of crystals when they are first observed. In this work, the proximity distance of 4 pixels was found to be adequate.

6.4.5 Manual Image Analysis

After the completion of a given scaling experiment, manual image analysis was conducted for selected captured images using the MSIA software for comparison with
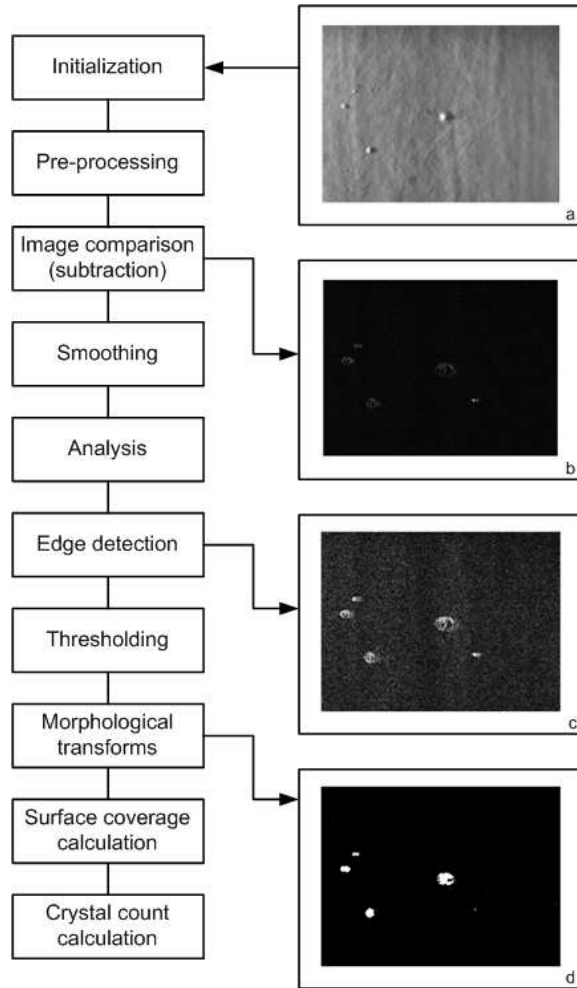
Figure 6.3: Flowchart of image analysis algorithm with representative example image outputs for selected algorithm steps: a) original camera image saved to MeMo computer disk, b) image resulting from subtraction of two most recently captured images (black pixels represent little to no change in pixel value and white denotes large changes when compared to the previous image), c) subtracted image after image filtering and edge detection, and d) final cumulative scaling image after morphological transforms.

131

the automated real-time image analysis software developed for the MeMo system. From the experimental data, the surface coverage was determined "manually" using digital image analysis software (Fovea plug-in for Adobe Photoshop) as discussed in [75]; each individual crystal in the images was outlined by hand and colored (e.g., red), then, built in functions were used to determine the surface area covered by these crystals, as well as the total number of crystals present in each image. Clearly, the above manual image analysis method would be intractable for an entire set of images; therefore, the "manual" image analysis method was only utilized for a fraction of the images from each run.

6.5 Results and Discussion

6.5.1 Mineral Scale Surface Coverage

The performance of the online surface scale image analysis illustrated in Fig. 6.4 and Figs. 6.5 - 6.7 for the scaling runs without and with antiscalant addition, respectively. The percent of surface coverage by mineral scale as determined by online automated image analysis follows the manual image analysis reasonably well, with a maximum deviation of about 3%. Agreement between the manual and automated image analysis was excellent at the early stages of scale development (when less than about 10% of the surface in the monitored area was covered by scale). These early stages of detection are most critical since scale mitigation actions (e.g., increasing antiscalant

dose, initiation of membrane cleaning or adjustment of operating conditions) would most likely be desired early in the membrane scaling process [86]. Although membrane images were analyzed until membrane surface coverage reached approximately 15-25%, at this high level of surface scale buildup significant flux decline would be expected when monitoring the tail membrane element of RO plants [47] which could possibly lead to membrane damage. The rate of surface mineral scaling decreased with the application of 3 ppm of antiscalant AS2 (run 4, Fig. 6.7) with scale coverage of 15% reached in 260 h (image 29). In contrast, doses of 1.5 ppm AS1 (run 2) and 3 ppm AS1 (run 3) were insufficient to retard gypsum scaling with 15% surface coverage reached within 7.25 h and 5.25 h, respectively, similar to the run without antiscalant dosing. The above results demonstrate that, in addition to monitoring mineral scaling, the MeMo system can also be used to assess antiscalant effectiveness in suppressing mineral scaling.

The crystal size and shapes can vary depending on the antiscalant type and dose (Table 6.2) as illustrated by the image insets in Figs. 6.4 - 6.7. Also, given the three-dimensional nature of surface crystals, the ability to detect crystal edges may be influenced by shadows and effect of neighboring crystals. Notwithstanding, in the present work a single set of UT and LT settings for a given scaling experiment were found sufficient to provide a reasonable level of mineral scale detection accuracy. It is acknowledged, however, that for a broader range of application of the MeMo system the hysteresis threshold tolerances (UT and LT, Sec. 6.4.3) may have to

be adjusted over the course of monitoring, via automated image calibration, for the specific lighting conditions and type of surface crystal topography.

Early scale detection, while avoiding false positive crystal detection, prior to the observation of measurable permeate flux decline was demonstrated for the two scaling experiments with (run 3, 3 ppm AS1) and without (run 1) antiscalant addition. Membrane surface monitoring in the MeMo cell (in the region near the channel exit) revealed significant scale (Figs. 6.4, 6.6), before any measurable flux decline (Fig. 6.8). It can also be seen that the surface coverage and crystal counts for runs 1 and 3 (Figs. 6.4, 6.6, 6.9, and 6.11) track very closely with the manual analysis, demonstrating the avoidance of false positive detection. Even though the surface coverage analysis is accurate, depending on the feed solution characteristics and the type of additives present, it may be desirable to monitor the number of crystals present on the membrane surface since early detection of mineral crystals is even more pronounced via the monitored crystal count.

6.5.2   Crystal Identification and Crystal Count

Crystal count was achieved once crystals were identified as described in Sec. 6.4. It is important to recognize that identification of the first crystal was achieved (Figs. 6.9, 6.11) before the detection of any measurable flux decline (Fig. 6.8). The first crystals are detected (in runs 1 and 3) between 60 and 75 minutes after the start of the experiment, while the permeate flux is still between 96-100% of the original
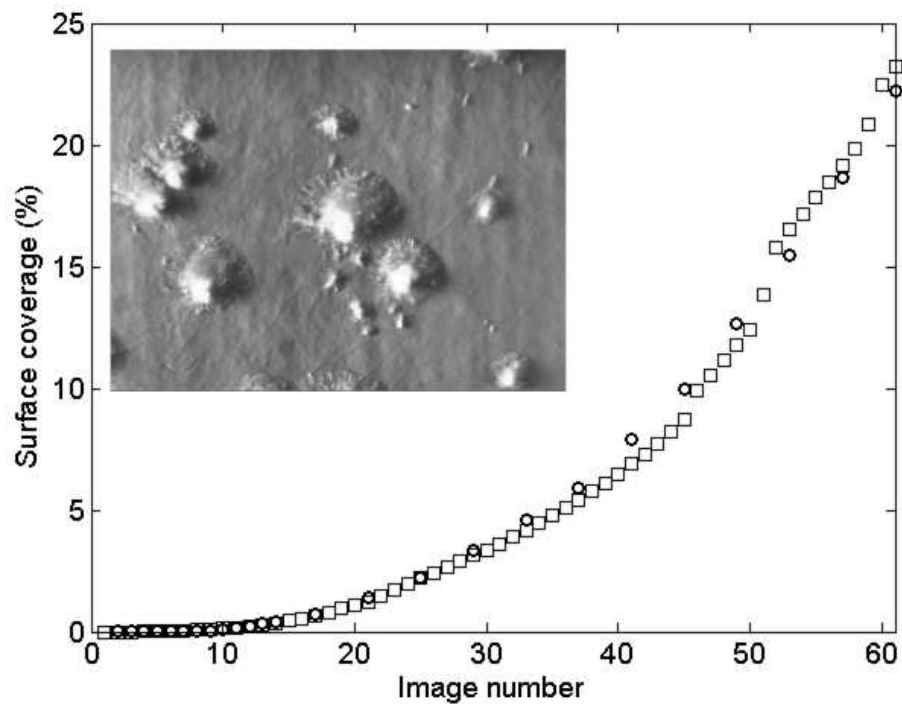
Figure 6.4: Percentage of mineral scaled membrane area (in the MeMo monitored membrane area) obtained via "manual" image analysis (circles) and MeMo automated scale detection software (squares) for Run 1 (without antiscalant addition; Table 6.2) with threshold tolerances UT = 0.54, LT = 0.59, and minimum crystal size of 350 pixels. Inset shows membrane surface image 61.
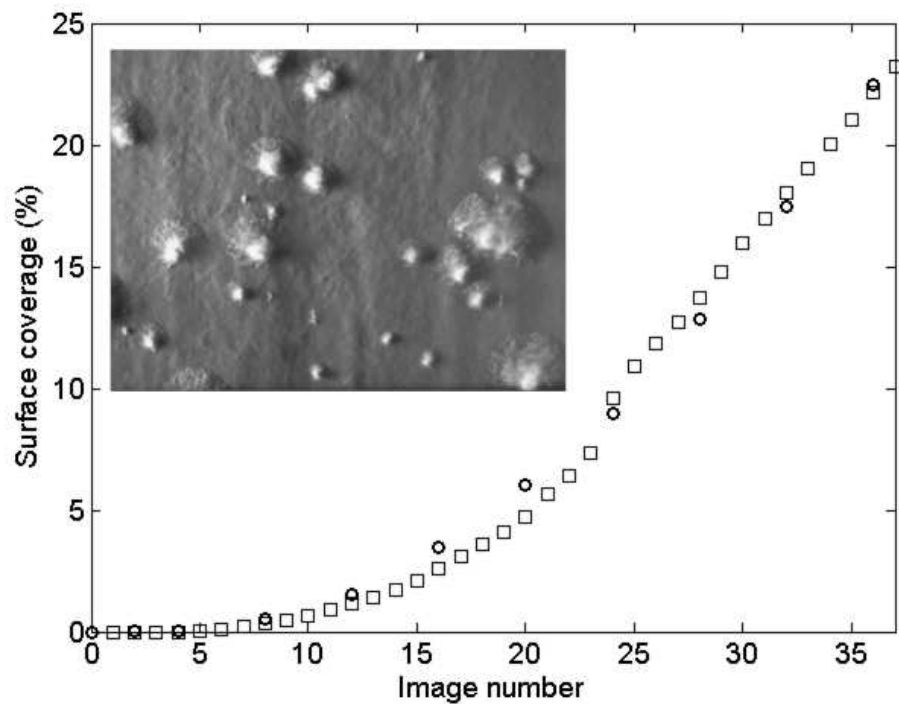
Figure 6.5: Percentage of mineral scaled membrane area (in the MeMo monitored membrane area) obtained via "manual" image analysis (circles) and MeMo automated scale detection software (squares) for Run 2 (1.5 ppm of AS1; Table 6.2) with threshold tolerances UT = 0.42, LT = 0.43, and minimum crystal size of 350 pixels. Inset shows membrane surface image 37.
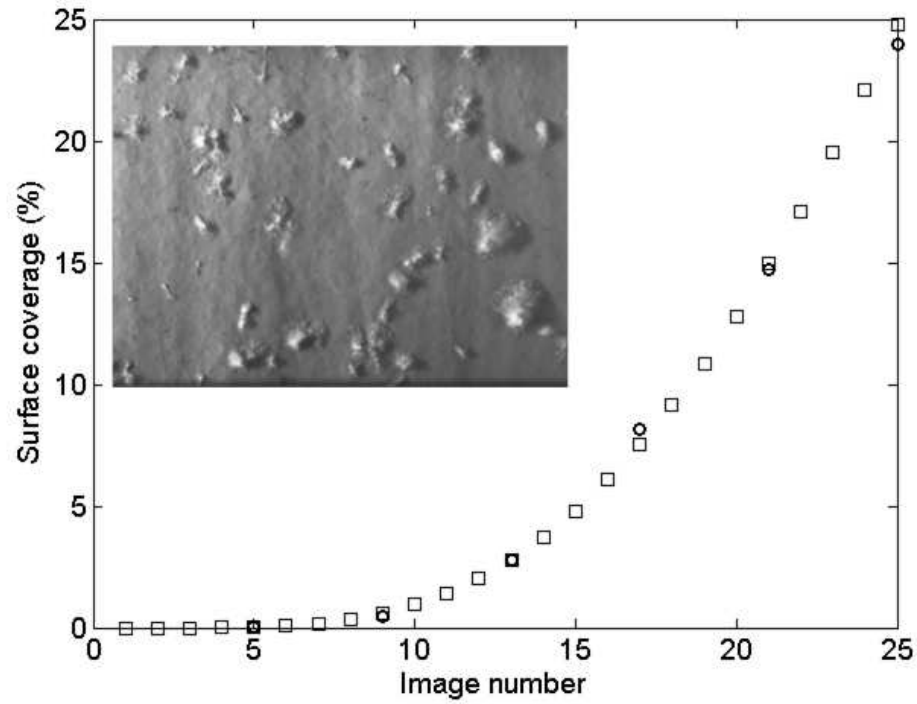
Figure 6.6: Percentage of mineral scaled membrane area (in the MeMo monitored membrane area) obtained via "manual" image analysis (circles) and MeMo automated scale detection software (squares) for Run 3 (3 ppm of AS1; Table 6.2) with threshold tolerances UT = 0.36, LT = 0.49, and minimum crystal size of 175 pixels. Inset shows membrane surface image 25.
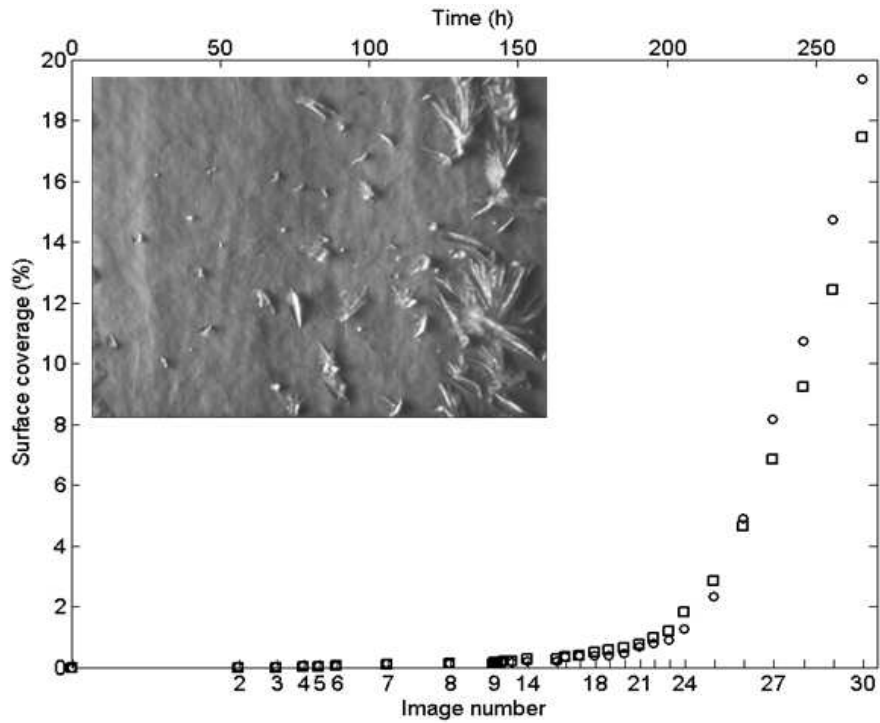
Figure 6.7: Percentage of mineral scaled membrane area (in the MeMo monitored membrane area) obtained via "manual" image analysis (circles) and MeMo automated scale detection software (squares) for Run 4 (3 ppm of AS2; Table 6.2) with threshold tolerances UT = 0.68, LT = 0.71, and minimum crystal size of 150 pixels. Inset shows membrane surface image 30.
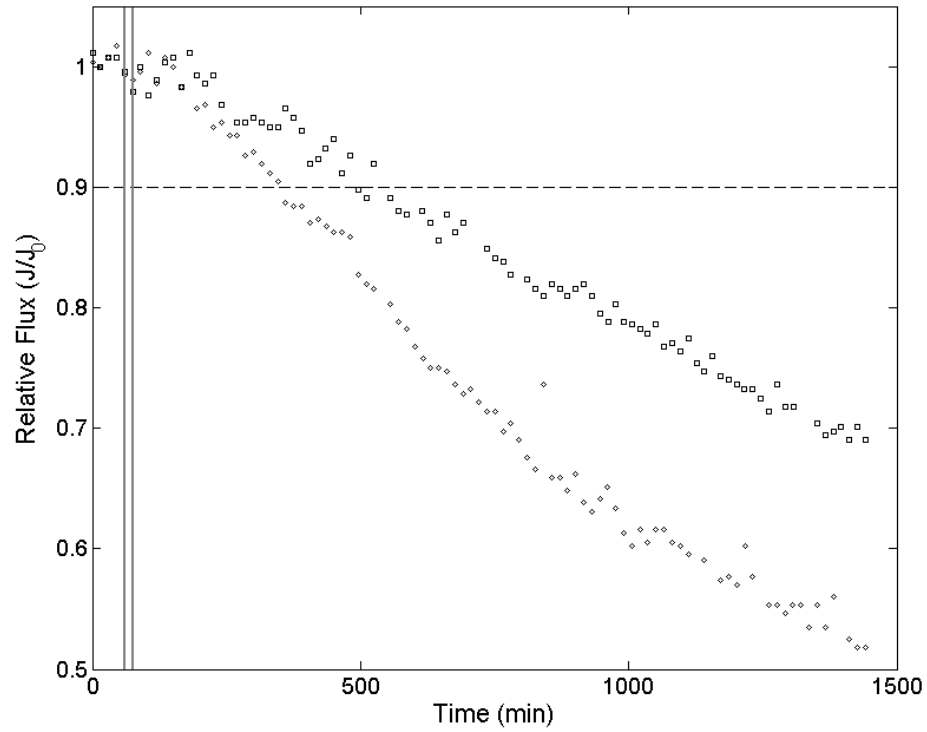
Figure 6.8: Relative permeate flux vs. time for run 1 (squares; without antiscalant) and run 3 (diamonds; 3 ppm AS1). The dashed horizontal line represents a 10% decline in permeate flux from the original flux at the start of the scaling experiments. The vertical lines represent the first detection of mineral salt scaling by the MSIA for runs 1 and 3 (at 60 and 75 minutes, respectively).

flux. By the time the permeate flux has decreased to measurable levels (10% flux decline), multiple crystals are already visible on the membrane surface. The crystal count also shows good agreement with the "manually" determined crystal count, with a maximum deviation of about 6 crystals between the MSIA result and the "manual" count. It is also noted that this maximum deviation occurs when many crystals are present on the membrane surface; the MSIA crystal count is much more accurate at the beginning of the scaling process when critical control decisions must be made (e.g., to activate a cleaning process or change operating conditions to avert flux decline).

Scale monitoring using the MeMo device can be particularly useful for triggering RO scale mitigation actions as demonstrated for the present system for RO plant operation in feed flow reversal mode [86]. In such applications, appropriate automation is required to enable adjustment of the pressure and flow rate in the MeMo RO cell so as to match the condition of solution supersaturation at the membrane surface to that in the RO plant element being monitored [86]. The MeMo device can also be used in a stand-alone mode (i.e., prior to its connection as an online detector) to determine the operational parameters that lead to scaling ([86], [85], [74]). Such information can provide knowledge of the operational region(s) that would result in mineral scaling and also allow one to arrive at optimal image analysis settings to enhance scale detection.
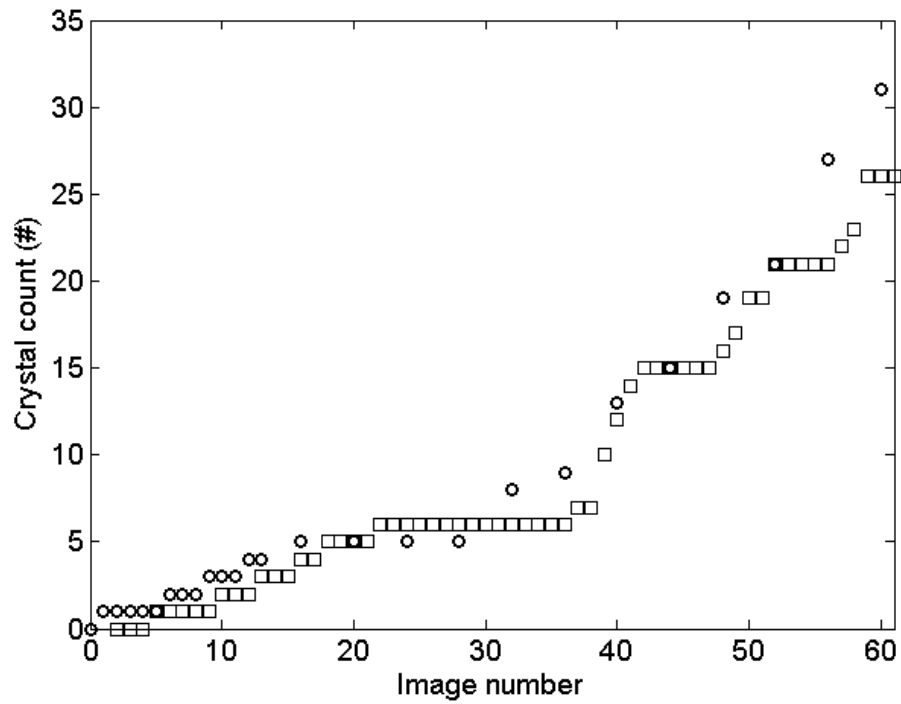
Figure 6.9: Crystal count in the MeMo monitored membrane area obtained from both "manual" (circles) and automated scale detection (squares) for Run 1 (without antiscalant addition; Table 6.2) where UT = 0.54, LT=0.59, and minimum crystal size of 350 pixels.
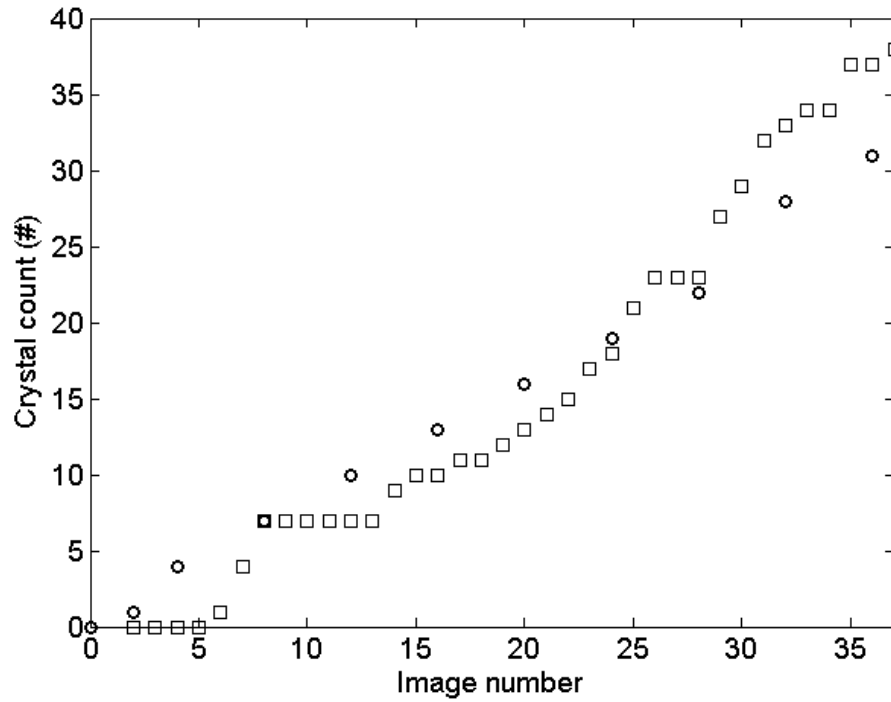
Figure 6.10: Crystal count in the MeMo monitored membrane area obtained from both "manual" (circles) and automated scale detection (squares) for Run 2 (1.5 ppm antiscalant AS1; Table 6.2) where UT = 0.42, LT = 0.43, and minimum crystal size of 350 pixels.
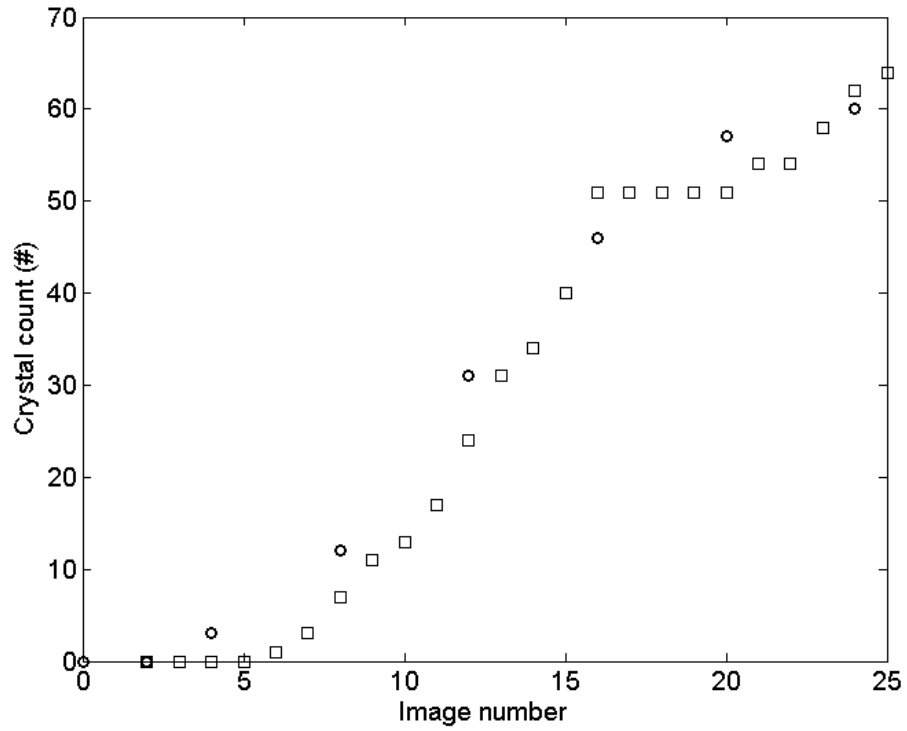
Figure 6.11: Crystal count in the MeMo monitored membrane area obtained from both "manual" (circles) and automated scale detection (squares) for Run 3 (3 ppm antiscalant AS1; Table 6.2) where UT = 0.36, LT = 0.49, and minimum crystal size of 175 pixels.
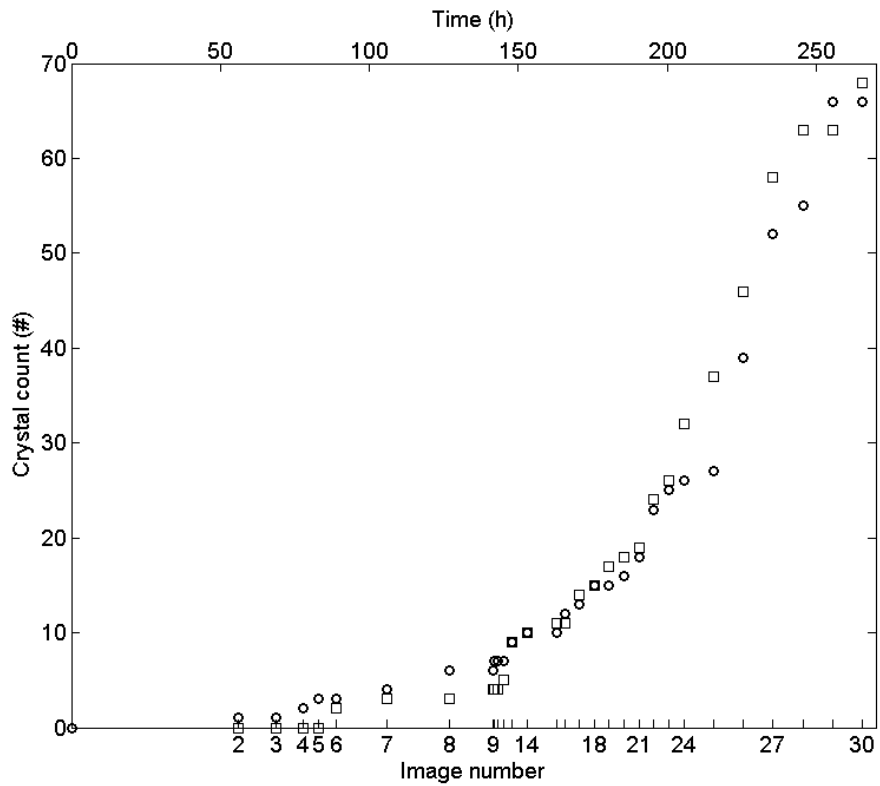
Figure 6.12: Crystal count in the MeMo monitored membrane area obtained from both "manual" (circles) and automated scale detection (squares) for Run 4 (3 ppm antiscalant AS2; Table 6.2) where UT = 0.68, LT= 0.71, and minimum crystal size of 150 pixels.

6.6 Feed Flow Reversal

Using the accurate mineral scale detection provided by the MSIA software and MeMo detector to determine surface scale coverage fraction and number of crystals, it was desired to design a control strategy to mitigate RO membrane mineral scaling. A recent method of RO operation called feed flow reversal has been developed, preventing scale formation without the addition of expensive chemicals or extensive periods of system down-time [71]. In this approach, a system of solenoid valves is used around the membrane modules configured specifically so that the direction of the feed flow through the membrane units can be reversed (see Fig. 6.14). This reversal of the feed flow also reverses the axial salt concentration profile [40] at the surface of the membrane, thereby exposing the mineral salt scaled areas to an undersaturated feed (w.r.t. scalants) leading to dissolution of the surface mineral crystallization [71] as seen in the schematic of Fig. 6.14.

Initially, solenoid valves s_2 and s_3 are closed and s_1 and s_4 are open. When flow reversal is initiated, solenoid valves s_2 and s_3 are opened, followed by the closure of solenoid valves s_1 and s_4 to re-direct the feed water, making the tail RO element (in normal mode) now the feed element (in FFR mode). Additional details regarding the FFR process and the experimental set-up on the M3 experimental RO system can be found in [38, 39].
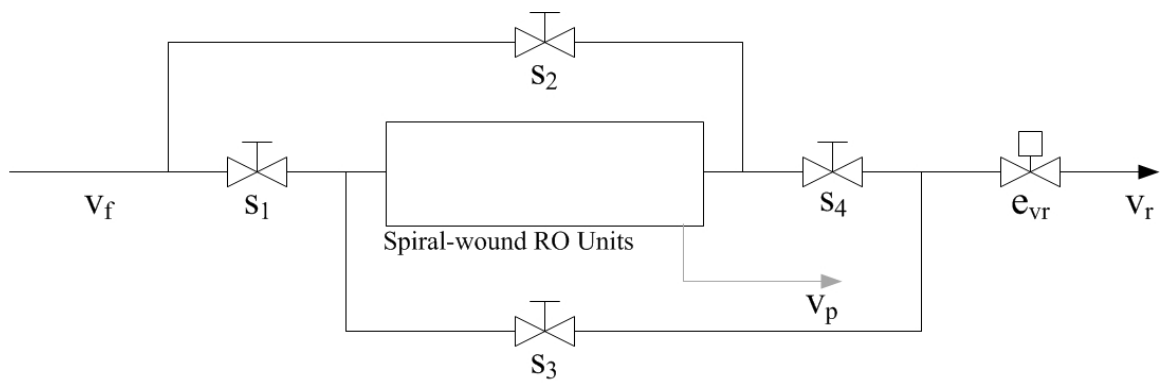
Figure 6.13: Schematic of RO system with flow reversal capability. s_1 - s_4 are solenoid valves used to switch between normal flow and feed flow reversal (FFR) modes; v_f, v_p and v_r denote the feed, permeate and retentate stream velocities, respectively.
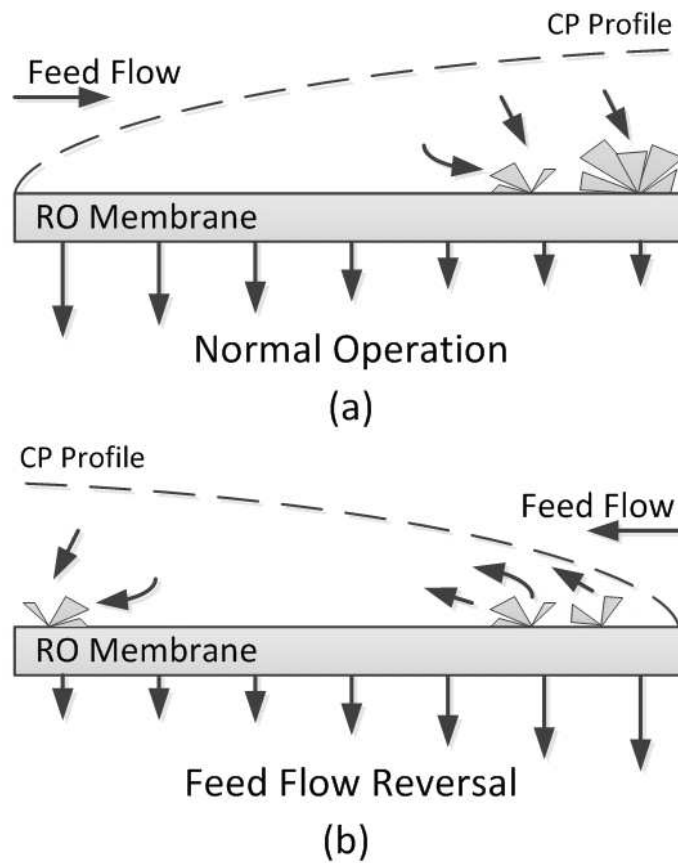
Figure 6.14: Schematic of the FFR process in the RO membrane feed channel. (a) shows the salt concentration and permeate flux profiles in normal flow mode with membrane crystallization (arrows toward existing crystals denote crystallization from bulk feed solution). (b) shows the salt concentration and permeate flux profiles in FFR mode with membrane crystal dissolution in the case of an undersaturated feed solution (arrows away from existing crystals denote crystal dissolution into bulk feed solution).

6.7 Automated Image Analysis for FFR Triggering Control

Using the FFR process, actuated by the real-time MSIA/MeMo membrane scaling detection, to mitigate mineral salt scaling was explored on two separate RO desalination systems. The first of these two systems was a brackish water RO system (BWRO) containing six RO elements in three pressure vessels (in series) at Ben Gurion University (BGU) in Israel. The MeMo detector was installed on the BWRO system, and the MSIA software was connected to the system's programmable logic controller (PLC). The MSIA software was set up to run on a local computer, and the signal for flow reversal actuation (3.3 VDC) was communicated through the laptop's parallel port pins to the PLC. When the MSIA software determined that the fractional surface coverage of scaling had reached the user-defined threshold (17% coverage in this work), the signal was transmitted to the PLC (via the parallel port) to actuate the system switch to feed flow reversal (FFR) mode. Additional details regarding the experimental conditions and system set-up can be found in [84, 83].

It is shown in Fig. 6.15 that the MeMo monitor correctly detects the onset mineral scaling on the membrane surface, since the permeate flow rate of the tail elements (in pressure vessel 3, "PV3") showed a decline due to mineral scale coverage. It was also shown that the flux through the membrane in the MeMo monitor is also
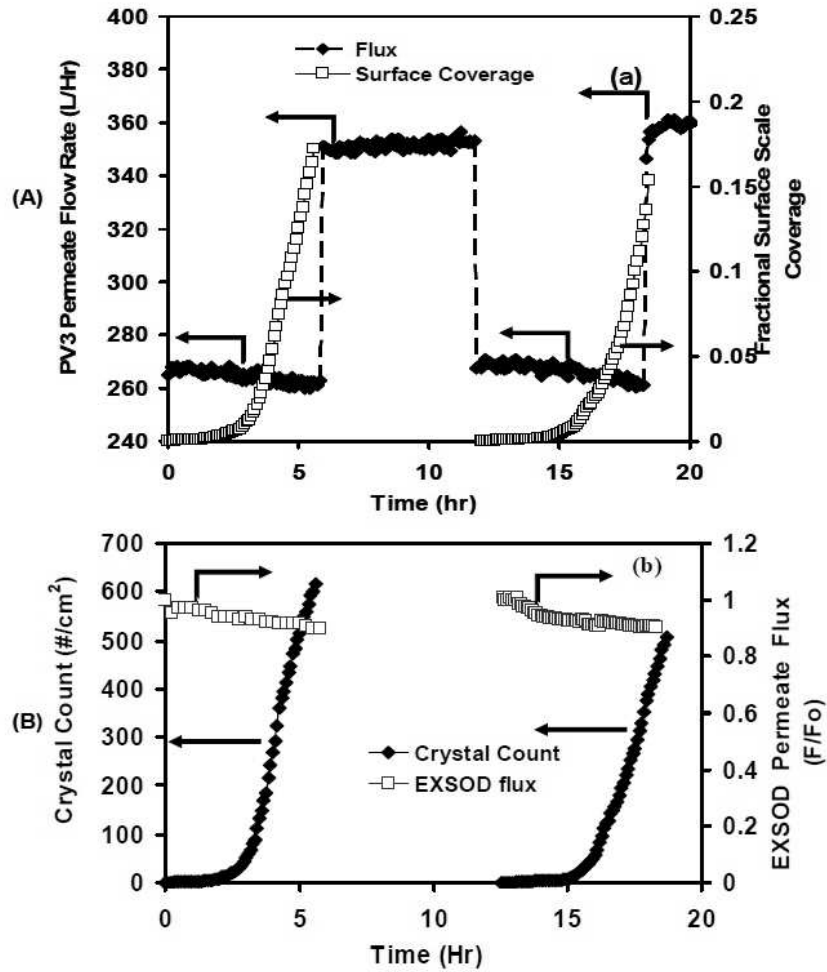
Figure 6.15: Demonstration of BWRO plant operation in normal feed flow and feed flow reversal modes in Ben-Gurion University study. (a) shows the permeate flow rate from the plant's tail RO elements (PV3) in conjunction with fractional scale coverage determined by MSIA software over the course of two FFR triggering cycles. (b) shows the permeate flux from the ex-situ membrane monitor and the number of crystals on the detector membrane surface as determined by the MSIA software [83]. For the results presented, feed pressure was in the range of 12-20 bar, and the feed flow rate was in the range of 800-1200 L/h.

declining due to the crystallization of the mineral salts on the membrane surface. After the MSIA software detects that the fractional coverage of scale has reached the pre-defined threshold, FFR was initiated, and the MeMo detector membrane was washed with RO system permeate water to remove the mineral salt scaling (to "reset" the surface coverage of the MeMo detector). After the FFR process was complete (in this study, FFR mode was conducted for the same duration as the previous normal flow mode with the addition of five minutes), the MSIA software signals the PLC to return the system to normal flow mode.

From these results, it is shown that the MeMo detector, paired with the MSIA software, was able to successfully detect the mineral scaling on the membrane surface and actuate the FFR process to remove the scaling. It was also demonstrated that the permeate flow rate in the tail elements (PV3) is recovered to its normal (pre-scaled) value, showing that the FFR process is successful in mitigating the effects of mineral salt scaling on the RO plant.

6.8 Automated Image Analysis for Multi-Cycle FFR Control

The feasibility of using the MSIA software with feed flow reversal control was later demonstrated over multiple cycles in the laboratory using the M3 system (set-up shown in Fig. 6.16). The MeMo monitor was attached to a side-stream of the
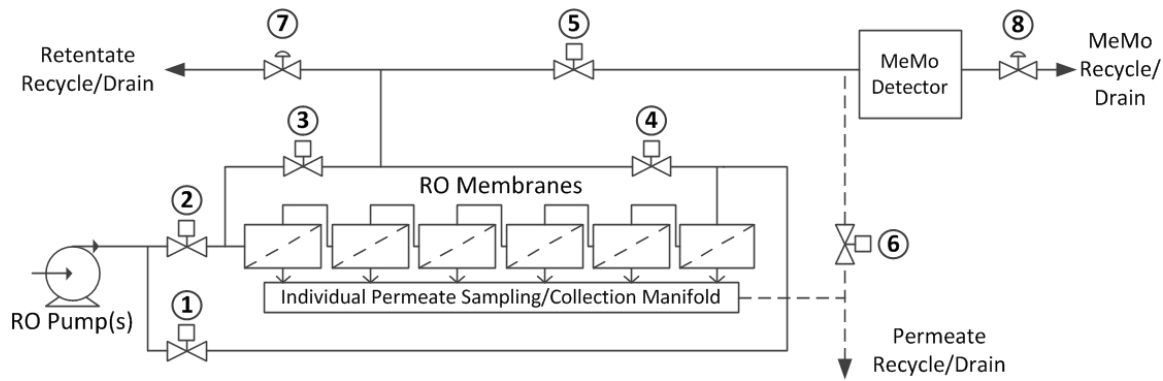
Figure 6.16: Schematic diagram of the M3/MeMo flow reversal system showing the location and arrangement of the actuated valves, pressure vessels and permeate collection network (dashed lines). Valves 1-4 are the M3 flow reversal solenoid valves, valves 5-6 control the feed to the MeMo detector (normal operation with high-pressure M3 retentate or detector cleaning with M3 permeate), and valves 7-8 are the actuated control valves for controlling M3/MeMo concentrate flow rate/system pressure.

pressurized retentate line, with additional solenoid valves for allowing cleaning of the MeMo cell with permeate (valves 5 and 6 in Fig. 6.16). A continuously-controlled needle valve (valve 8 in Fig. 6.16) was also used in order to adjust the scaling conditions (pressure, flow rates) in the MeMo cell. The M3 system was arranged to allow for feed flow reversal (FFR) operation (as described previously in Sec. 6.6), as well as individual membrane flux monitoring using a manifold of permeate sampling valves. This allowed for the lead and tail element permeate flow rates/permeate fluxes to be monitored over the course of the FFR experiment.

At the beginning of the experiment, the system was arranged in total recycle mode

and set to operate at high recovery (70% recovery) with the MSIA software fractional surface coverage threshold set to 50%. The feed solution was composed of calcium, sodium, sulfate, and chlorine ions, with gypsum ($CaSO_4$) being the scalant of most concern. For the experiment presented in this thesis, the feed gypsum saturation index was $SI_g^{feed} = 0.454$, and during automated operation, the saturation indexes of gypsum at the membrane surface for the M3 and the MeMo detector were 3.45 and 3.65, respectively (in the monitored/exit region of the membrane).

After the fractional surface coverage reached 50% in the MeMo detector/MSIA software, a signal was sent to the M3 system to switch to flow reversal mode (with the tail element becoming the lead element). This process was repeated over the course of multiple scaling/cleaning cycles to determine if the scaling process on the membrane surface could be reversed (and in turn, if the permeate flux of the tail RO membrane element could be recovered). In the flow reversal mode, the MeMo detector was cleaned with permeate from the M3 system in order to "reset" the fractional surface coverage in the detector. Figure 6.17 shows six flow reversal cycles from an 88 hour experiment.

From the experimental data, it was shown that the system was able to initiate multiple cycles of feed flow reversal and was able to recover the permeate flux (over 90% flux recovery after 15 cycles) lost due to mineral scale coverage of the membrane surface while operating without chemical additives (such as anti-scalants). These results demonstrate that the feed flow reversal process, in conjunction with the MSIA
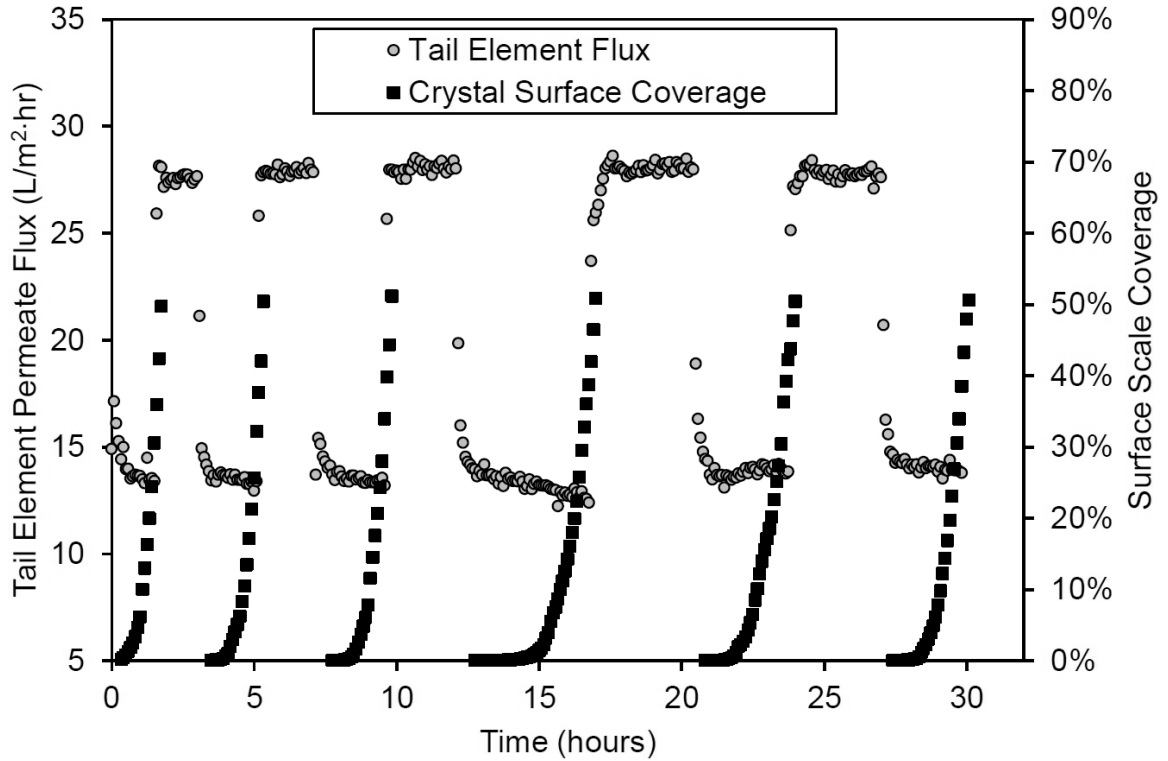
Figure 6.17: Normalized tail element permeate flux (gray circles) and percent surface scale coverage (black squares) as observed in MeMo for the first six cycles from an 88-hour experiment. The lower permeate flux curves designate the forward flow operation while the top permeate flux curves denote the flow reversal operation (tail element becomes lead element in FFR mode). For the results presented, feed pressure was set at 168 psi, and the feed flow rate was maintained at 2.25 gpm.

software as a trigger, allows for extended automated operation of a RO membrane desalination system with feed water of high scaling propensity. However, the feed SI of the mineral scalants must be below unity, otherwise dissolution of the mineral scale on the membrane surface by the feed water will not occur.

An additional benefit to this method is that this process is also highly scalable to large-scale RO desalination processes, requiring only the installation and control of the MeMo detector and flow reversal valves. Additional details regarding the experimental conditions and additional results can be found in [39].

6.9   Conclusions

An approach to real-time analysis of the formation of mineral scale on reverse osmosis (RO) membranes was developed using an ex-situ direct observation membrane monitor (MeMo). Real-time images of the membrane surface in the MeMo membrane channel were analyzed online to detect the onset of mineral crystals and to monitor the evolution of the fractional coverage by mineral salt crystals and crystal count. Image analysis software, which was developed specifically for the MeMo system, was capable of real-time detection of the formation and growth of mineral crystals on the membrane surface. The automated image analysis program (operating either on-line, or in a post-processing mode) was shown to accurately determine the membrane surface coverage by mineral salt scaling and the number of crystals present in the

observation area of the detector. Using the presented scale monitoring approach, a demonstration of the use of the MeMo/MSIA software in triggering a scale mitigation process called feed flow reversal (FFR) was conducted at Ben-Gurion University in Israel. From this study, it was shown that the MeMo/MSIA software was effective in triggering the FFR process in an automated fashion. It was also shown that the automated control of the feed flow reversal process was effective in maintaining RO membrane permeate flux under high recovery (and high scaling propensity) conditions in the BGU study as well as over the course of multiple FFR cycles conducted on the M3 RO system.

In addition to potential applications for RO plant monitoring, the presented monitoring system with its surface analysis software can serve to acquire information regarding mineral scale kinetics (e.g., rate of nucleation and rate of growth of individual crystals) and to evaluate the suitability of other scale measures (e.g., geometrical measures of crystal shape and crystal number density) for optimal control strategies [85, 57].

Chapter 7

Second Generation Experimental Ultrafiltration/Reverse Osmosis Water Desalination System

7.1   System Overview and Improvements on the First-Generation M3 System

Extensive experimental lab and field work with the first generation M3 RO system suggested a number of system design and operational strategies for improvement of RO system efficiency. Accordingly, a second-generation compact and modular RO system (CoM2RO) was developed to improve on the operability and versatility of the M3 system. With the new system design, several major aspects were added/improved with respect to the design of the first-generation system:

- The second-generation system is designed with a dedicated pre-filtration (MF/UF)

skid for comprehensive feed pretreatment, allowing for the treatment of a wider range of feed waters (seawater, agricultural drainage water, etc.) along with automated adaptive filtration/cleaning.

- The CoM2RO system is designed for integrated operation, which allows for operation with no intermediate tank between the MF/UF and RO processes, decreasing the overall system footprint.

- The capacity of the CoM2RO system is greatly increased over the feed/permeate capacity of the M3 system (feed water flow rate of up to 36 GPM compared to a feed flow rate of 8.3 GPM with the M3 system).

- The CoM2RO system is designed in a completely modular fashion to allow for the implementation and testing of novel components (membranes, filters, sensors, etc.) and system control strategies.

- The embedded computing hardware on the CoM2RO is much more powerful (three distributed embedded controllers as compared to one on the M3 system) and modular for increased computing power, robustness, and system/operator safety.

- A more extensive network of sensors and actuators is implemented on the CoM2RO system in order to provide more information about the system operating state, and also to provide additional data to use for system characterization/process control (e.g., the addition of turbidity and in-line temperature

measurements).

- The modular design (with regards to the system components and also the embedded control hardware) allows for the implementation of advanced software architecture (soft sensors, event-based control, "smart" operation) and novel control algorithms.

The CoM2RO system is also designed with the end-goal of shipboard deployment, necessitating the small system footprint (less than 340 ft^3), reduced energy usage, and highly automated ("smart") operation to minimize system maintenance, facilitate system cleaning, and minimize user interaction. A schematic of the general process components is shown in Fig. 7.1.
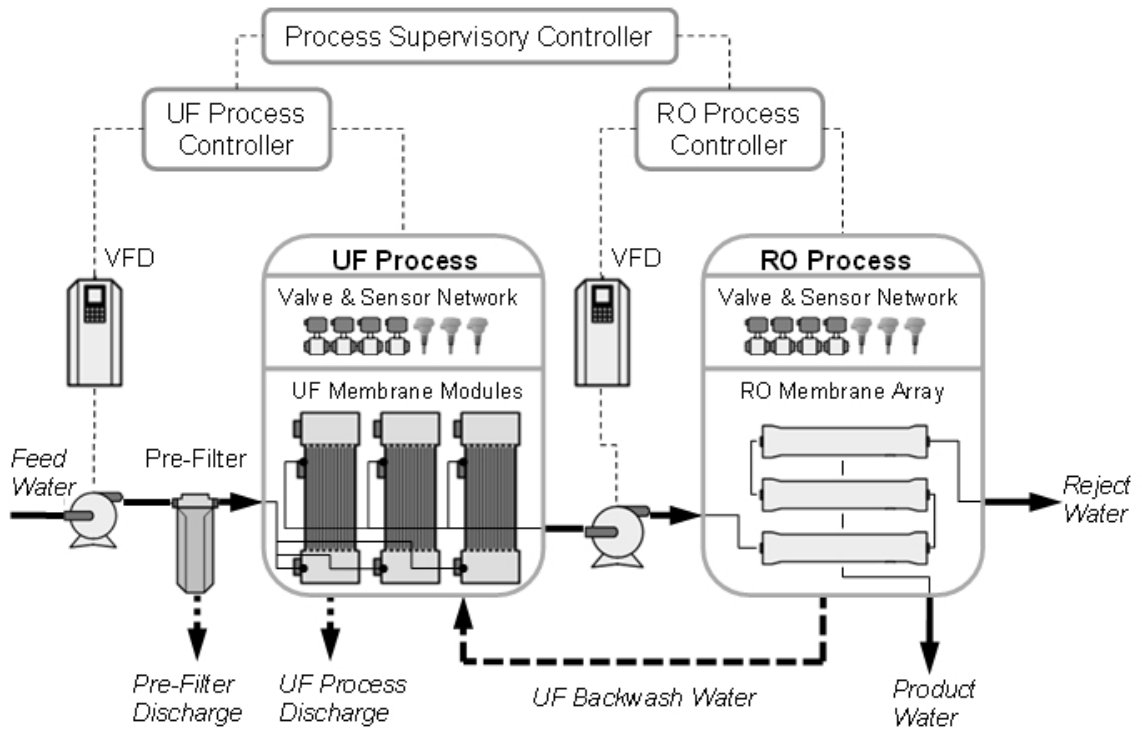
Figure 7.1: CoM2RO control system schematic.

7.1.1 Concepts and Design

Pre-filtration/UF Process

The CoM2RO system contains a prefiltration skid with three hollow-fiber (inside-out) ultra-filtration (UF) modules (Inge, Germany), a centrifugal low-pressure pump (with VFD control, 5hp, Price Pump, USA), a self-cleaning 200 μm filter (Amiad, Israel), and an extensive network of actuated 2-way and 3-way valves for flow path control. Multiple types of filtration and backwashing are possible with the CoM2RO system; the UF modules can operate in a dead-end filtration mode with water entering from the top or bottom, and can also operate in a cross-flow mode with water entering from the top or bottom. Additionally, the modules can be backwashed with the water entering from the permeate side exiting from either the top or the bottom. Small accumulators (2 x 5L) are located on the UF backwash line to provide an optional "shock" backwash (high-flux pulse through the hollow fibers to dislodge foulants). These configurations for forward filtration/backwash can be customized with respect to timing, pressure, and flow rates, and can also be initiated in various orders/sequences. The UF system also contains extensive sensing and monitoring capabilities (flow, pressure, turbidity, pH, ORP, temperature) to assist the control system and the user in making operational decisions. Combining the sensor readings with calculations conducted on a supervisory embedded controller allows the system to automatically make decisions regarding the system flow configuration (e.g., adap-

tive backwash based on degree of flux decline or trans-membrane pressure increase) and implement these decisions in real time.

Another novel aspect of the CoM2RO system is the option to conduct integrated operation (UF/RO) without an intermediate tank between the pre-filtration and RO processes. Typically, an intermediate tank is used to collect UF permeate (or other types of pre-treatment effluent) before the water is sent to the RO process for membrane-based water desalination. This intermediate tank decouples the dynamics of the system, making overall system control much easier. However, the addition of this intermediate tank drastically increases the footprint of the system. Through the advanced ("smart") distributed control present on the CoM2RO system, the need for an intermediate tank is eliminated.

RO Process

After the feed water has been filtered through the UF modules, it flows to the RO process skid where a safety pre-filter is installed before the high-pressure axial piston positive displacement pump (with VFD control, 30hp, Danfoss, Denmark). This pre-filter is present in order to protect the RO pump from particulate matter in the event of a breach in the UF module integrity. The feed water is pressurized to the required pressure (depending on feed water salinity and type of RO membranes) and fed into the RO pressure vessels containing the spiral-wound membrane modules (seawater membranes, 8 in. diameter, 40 in. length, Dow, USA). The initial CoM2RO

system construction contains three seawater spiral-wound membranes in series for achieving system recoveries of up to approximately 40%. Additional frames can be added in a modular fashion to expand the RO processing capacity (or increase system recovery, if low-salinity feed water is used). After passing through the RO membrane vessels, the concentrate stream is throttled through an actuated needle valve (Jordan Valves, USA) which is controlled through the distributed control system to adjust system pressure/recovery/flow rates. The concentrate stream can then be discarded, or used as a backwash water source for the UF membranes. The RO permeate can be collected in an external permeate tank (if the water quality is sufficient for potable use), discarded as waste, or used as a backwash water source for the UF process. As in the case of the UF process, the RO skid also contains extensive sensing/monitoring (flow, pressure, pH, ORP, temperature, conductivity) as part of the local/distributed control system.

General Design Considerations and Construction

The system was designed on multiple frames to allow for easy assembly, transportation, and disassembly. Each frame was sized in order to fit through a narrow ship hatch door. The arrangement of frames also allows for quick and easy system augmentation with additional frames, such as additional RO membrane modules, or the MeMo ex-situ membrane monitor. The initial system design contains five frames; two frames for the ultra-filtration (UF) modules attached to one frame for the main UF

system piping and electronics, and one frame containing the RO high-pressure pump and membrane vessels, attached to the main RO frame containing the majority of the RO piping and electronics.

Mechanical construction of the CoM2RO system was conducted over the course of four months, followed by component testing and electrical wiring. The system was then moved to UCLA's Co-generation (power/water) facility for more extensive system-level testing. Additional details regarding system components are provided in Appendix B .

7.1.2 System Operability

The CoM2RO system was designed in order to accomplish several goals, including (but not limited to) minimization of system footprint, optional removal of an intermediate tank between pre-treatment (MF/UF) and RO processes, minimization of chemical usage, minimization of system energy consumption, the ability to adapt to changing source water, and "smart" local/remote automated process control for decreased human interaction and maintenance. In order to meet these goals, several novel designs were developed:

- Novel pre-treatment flow system with multiple flow options for various backwashing strategies.

- Novel component arrangement and framing for minimal system footprint.

- Electrically actuated components (2-way and 3-way valves, actuated retentate valve (continuous), variable frequency drives (VFDs), filters) for automated local/remote process control.

- Extensive sensor network with electrical feedback (4-20mA, relays, etc.) for automated sensing/data-logging/control.

- Distributed control architecture to allow for local/remote system operation as stand-alone UF, stand-alone RO, or integrated UF/RO system.

- Novel software design for data handling and process logic (automated decision-making), allowing for advanced adaptive backwashing strategies and "smart" system operation.

- Local and remote graphical user interfaces (GUIs) for user input (production rate, fault handling, etc.), as well as local/remote process control and monitoring.

- Modular hardware and software design for future augmentation and upgrading (addition of more detailed process models, additional sensors/actuators).

This dissertation includes a discussion of the concepts and implementation of the control hardware and software; a detailed discussion of process flow paths, process sizing/flow component selection (pumps/piping), as well as the detailed discussion of system energy optimization, logic-based decision making, and adaptive backwashing

strategies will be found in the dissertations of the other doctoral students working on the project.

Sensor/Actuator Selection

System components were selected with the above concepts as a basis (see Appendix B for details regarding component manufacturers and model numbers). In terms of the control system operability, the system was constructed using actuated 2-way/3-way valves that could be actuated through a relay module on the distributed control system. The valves selected required 12/24VDC for power; a common actuation voltage that is safer to work with than 115VAC. The valves are also equipped with limit switches (dry contacts for both open/closed positions on the GF Signet and Plast-o-matic valves, wet contact for the open position on the KZ valves) so that a signal can be sent to the digital input modules on the distributed control system to determine what position the valves are in (open/closed). This is necessary for the purposes of personnel safety, system reliability, and for fault detection/fault tolerant control algorithms. The self-cleaning pre-filter was also chosen due to the ability to control the cleaning cycle from the relay modules of the distributed control system. This aspect of the process control is important because the feed flow can drop sharply (depending on the system pressures) when the pre-filter cleans itself. This drop in flow/pressure could have adverse effects on the filtration/backwashing processes if the cleaning cycle is not timed appropriately.

The continuous actuators in the CoM2RO system (VFDs, actuated retentate valve, metering pumps) were also selected due to their ability to be controlled with a 4-20mA analog signal from the analog output modules on the distributed control system. The VFDs also have several dry relay outputs that can alert the control system/process operator to a number of important alarms/warnings that can occur in the VFD/motor/pump (the relay states can be read through the digital input modules). An extensive sensor network was also designed and constructed based on equipment that can output 4-20mA analog signals (proportional to the measurement). This is also a standard type of sensor output commonly used in industrial systems due to the fact that a 4-20mA signal will not degrade as much as a voltage signal over longer wires (voltage will drop slightly due to wire resistance). These 4-20mA signals from the sensors (pressure, flow, temperature, conductivity, pH, ORP, turbidity, etc.) can be read by the analog input modules on the distributed control system.

Control System Hardware Selection

The distributed control system hardware (National Instruments) was selected to allow for collection of all sensor data, while sending the appropriate control signals to the system actuators. Also, the controllers must contain the processing power (CPU) and memory (RAM) to conduct any type of data processing, decision logic, data communication, and control algorithms at high speed (0.5-10 Hz). Similarly, the controllers on the UF and RO frames must be able to log the system data and save it to a local

disk (USB drive for backup/emergency data logging). The selected controllers also followed the general modular approach, with detachable input/output (I/O) modules which perform multiple functions such as analog input (4-20mA), analog output (4-20mA), digital input/output (24VDC), and switching (relay outputs). Additional I/O modules can be added to the controller chassis (depending on the number of open slots) at any time. Additional details regarding system control components are also given in Appendix B .

Corresponding distributed control system software was designed and coded using LabVIEW [2]. At the lowest levels, the software utilizes a field-programmable gate array (FPGA) to communicate with the analog/digital I/O modules in the chassis, then sends the sensor data to the real-time controller (RTC) (an embedded version of LabVIEW running in a dedicated operating system without the need for a human-machine interface (HMI)). The real-time controller also has a dedicated CPU and memory for running more complex algorithms than the FPGA. The RTC is responsible for all calculation and data handling that does not occur on the supervisory PC. These functions include (but are not limited to): data logging, decision logic, data processing, checking hardware connectivity, safety sub-routines, FPGA communication, web-interface, and network communication. Two FPGA/RTC controllers are utilized for lower-level logic and control (one on each of the frames, UF/RO), and the remaining FPGA/RTC controller is used as a supervisory controller to coordinate actions between the lower-level controllers. The supervisory controller was also selected

with more RAM and CPU power so that it can coordinate the lower-level controllers as well as communicate with the graphical user interface (GUI) and supervisory PC. The distributed control hardware/software architecture (comprised of three separate real-time controllers/FPGAs) is described in detail in Sec. 7.3.

System PC Selection

A fanless, compact, touch-panel PC is also used as the local HMI for communication with the supervisory controller (and in turn, the lower-level controllers on the UF and RO frames). This computer was selected to be fanless so that the supervisory control enclosure (containing the unmanaged (no configuration interface or options) network switch, the wireless router, supervisory controller, and supervisory PC) would be waterproof. It should be noted that all electrical and mechanical components have been selected on the basis of their NEMA 4X (or higher) rating; this denotes that the component can be subjected to a low-pressure "washdown", and is suitable for outdoor (all-weather) use. The supervisory PC runs the GUI that interfaces with the supervisory controller, allowing the operator to set important process parameters (such as the desired production rate). The supervisory PC can also be used to interface algorithms running in other software or other locations (e.g., MATLAB [72], C, remote applications) with the LabVIEW GUI in a modular fashion. In this way, the operator can test alternative control strategies or augment the existing control strategies with additional process modeling.

168

7.2 Power Distribution and Electrical Systems

The power distribution and electrical systems for the CoM2RO system were also designed and constructed at UCLA. The electrical systems are housed in four enclosures (two on the UF skid, one on the RO skid, and one supervisory box that can be mounted on either frame). The external power (480VAC, 3-phase, 50A total and 115VAC, 1-phase, 25A total) is supplied by the site, and connected to the CoM2RO through separate heavy-duty insulated multi-conductor cables. The source power (both 480V and 115V) enters the system at the back through small plastic enclosures. The 480VAC, 3-phase power is supplied directly to the RO skid, and then is split into two paths (one through the RO contactor before the VFD, and one to the UF skid/contactor/VFD). The 115VAC power enters each individual skid through a heavy-duty insulated cable. A representation of the CoM2RO power distribution is shown in Fig. 7.2.

UF Electrical Systems

The UF skid has an enclosure containing the power supplies/ground fault circuit interruption (GFCI)/main fusing (converting 115VAC to 24VDC for sensors/valves, and to 12VDC for the KZ valves), and an enclosure containing the sensor/actuator connections, auxiliary fusing, valve relays, emergency safety relay, and the NI cRIO data acquisition (DAQ) system.
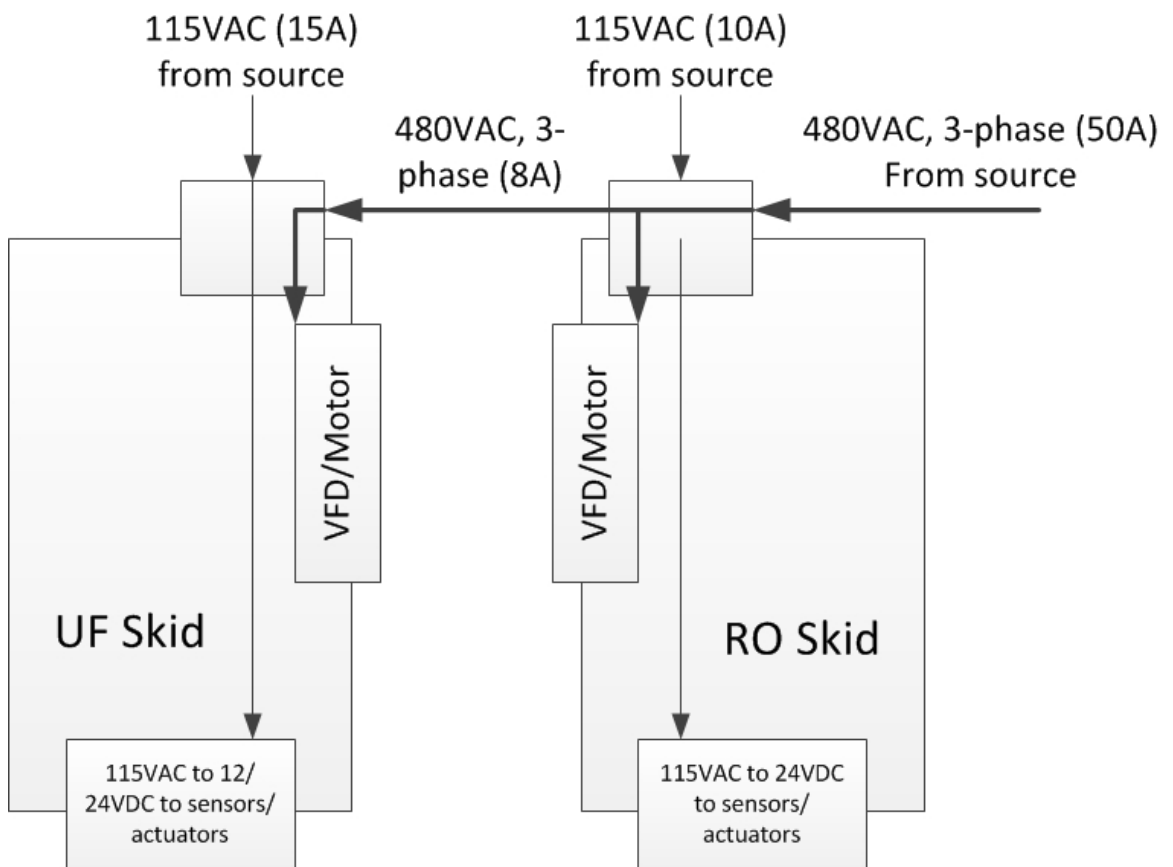
Figure 7.2: CoM2RO power distribution.

RO Electrical Systems

The RO skid has one enclosure which contains all of the same previously mentioned components as the UF skid (only one enclosure is necessary due to the lower number of actuators on the RO process).

Supervisory Enclosure Electrical Systems

The supervisory enclosure contains the supervisory NI cRIO DAQ system, the supervisory PC (touch-panel fanless PC), an unmanaged (no DHCP capabilities) ethernet switch (National Instruments, USA), all required fusing, and a power supply (powering the cRIO, PC, switch and enclosure lights/buttons). The supervisory enclosure can be mounted on either the UF or RO skid, and both distributed cRIOs (UF/RO) are connected to the supervisory enclosure through ethernet connections. The supervisory cRIO and supervisory PC are also connected to the unmanaged ethernet switch, creating a system-wide network for data transfer and distributed control.

General Safety Systems

Safety systems have also been incorporated, using emergency safety relays to de-energize the source power (480VAC, 3-phase) contactors for the UF and RO VFDs/motors. The safety relays will only allow the VFDs/motors to function if the high and low pressure switches are closed, and when the external emergency stop buttons are not triggered. The safety relays on both skids are interconnected such that if one emer-

gency stop button is pressed, the motors on both skids will stop (both safety relays are tripped). The sensors/controllers will still function in the event of an emergency stop event; this allows the control hardware to alert the user (locally or remotely) that an emergency condition has occurred. If a high/low pressure switch triggers the emergency stop, the controller can be locally or remotely power cycled in order to re-set the safety relay (provided that the condition that triggered the high/low pressure switch is fixed). If one of the emergency stop buttons is pressed, it must be manually re-set in order to operate the system.

Wiring diagrams for the electronics enclosures and additional information regarding CoM2RO system sensors/actuators can be found in the thesis addendum.

7.3   Control System Architecture

The CoM2RO system is operated through the use of a distributed control system comprised of three embedded control systems (one on each of the UF and RO frames, and one in the supervisory enclosure). Each of these controllers performs specific functions related to its position in the hardware and software architecture. These controllers are responsible for collecting all sensor/limit switch data, processing the data collected, performing complex logic-based operations, conducting process control calculations, sending signals to the process actuators, communicating to the other controllers on the local network, and other additional tasks. The appropriate divi-

sion of responsibilities and processing tasks among the controllers is essential to safe, reliable, robust, and efficient operation of the integrated UF/RO process. This architecture must address the required tasks, as well as optimize the usage of controller resources (CPU, RAM, hard disk space), while operating at a given sampling rate and communicating with local/remote HMIs.

7.3.1 Control System Hardware

Figure 7.3 shows the schematic representation of the major components in the control system network. As previously described, an embedded controller (FPGA/RTC) is located in the electrical enclosure box on each of the two frames (UF/RO). Two frames/controllers are used instead of a single frame/controller to allow each of the RO/UF processes to be operated in a stand-alone fashion. In this way, one can operate the RO with feed water from a different pre-treatment source; also, one can operate the UF process without the RO unit. The supervisory/HMI panel is removable, and can be mounted on either of the process frames. As shown in Fig. 7.3, all of the sensors and actuators on the frame are connected to the embedded control system through the electrical enclosure, which then directs the sensor/actuator signals to/from the I/O modules in the embedded controller chassis. The power conversion (from 115VAC to 12/24VDC) also takes place within the electrical system of each skid (contained in the electrical enclosure).

Once signals from the sensors are processed by the local embedded controller, the

173

resulting data is sent (via ethernet cable) to the supervisory controller located in the supervisory control enclosure (control system layout depicted in Fig. 7.4). The supervisory enclosure contains an ethernet switch, which directs the data from the local controllers to the supervisory embedded controller or the supervisory PC (or both). The remote internet connection and wireless router are also plugged into the ethernet switch with ethernet cables. This supervisory enclosure also contains the supervisory PC.

Figure 7.4 shows the physical connections between the different hardware targets in the system. In each embedded controller, the sensor/actuator wires are physically connected to the FPGA through the I/O modules. The FPGA is connected to the RTC, which then is directly connected to the unmanaged switch through the ethernet cable. The supervisory enclosure functions in a similar fashion, and also includes buttons and LEDs mounted to the front of the enclosure which enable user input/control in the event of supervisory PC failure. These buttons and LEDs are directly connected to the I/O modules in the supervisory controller chassis. The "C" on some of the hardware targets denotes a unique code that is deployed on the hardware target.
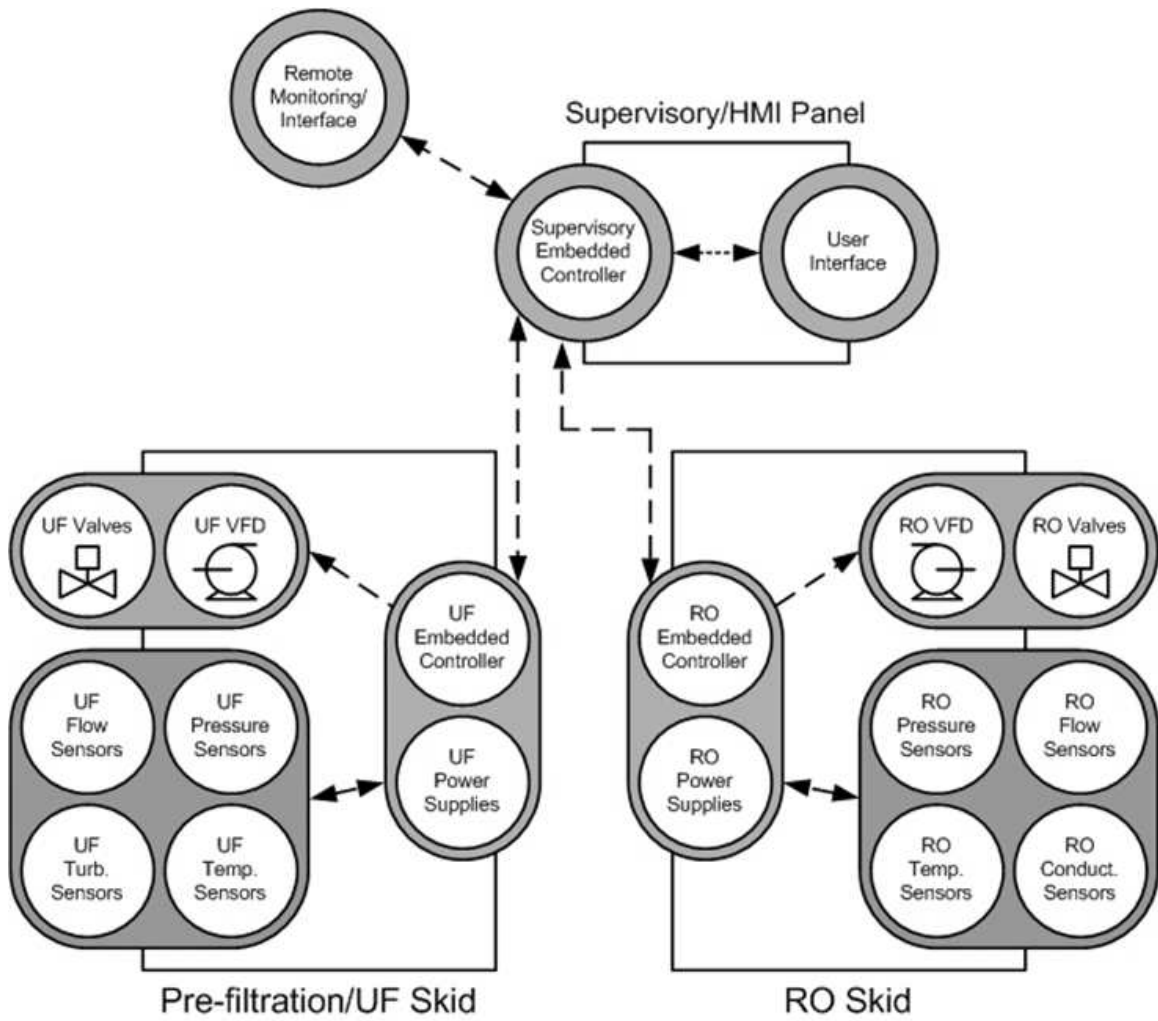
Figure 7.3: Schematic representation of the CoM2RO distributed control system.
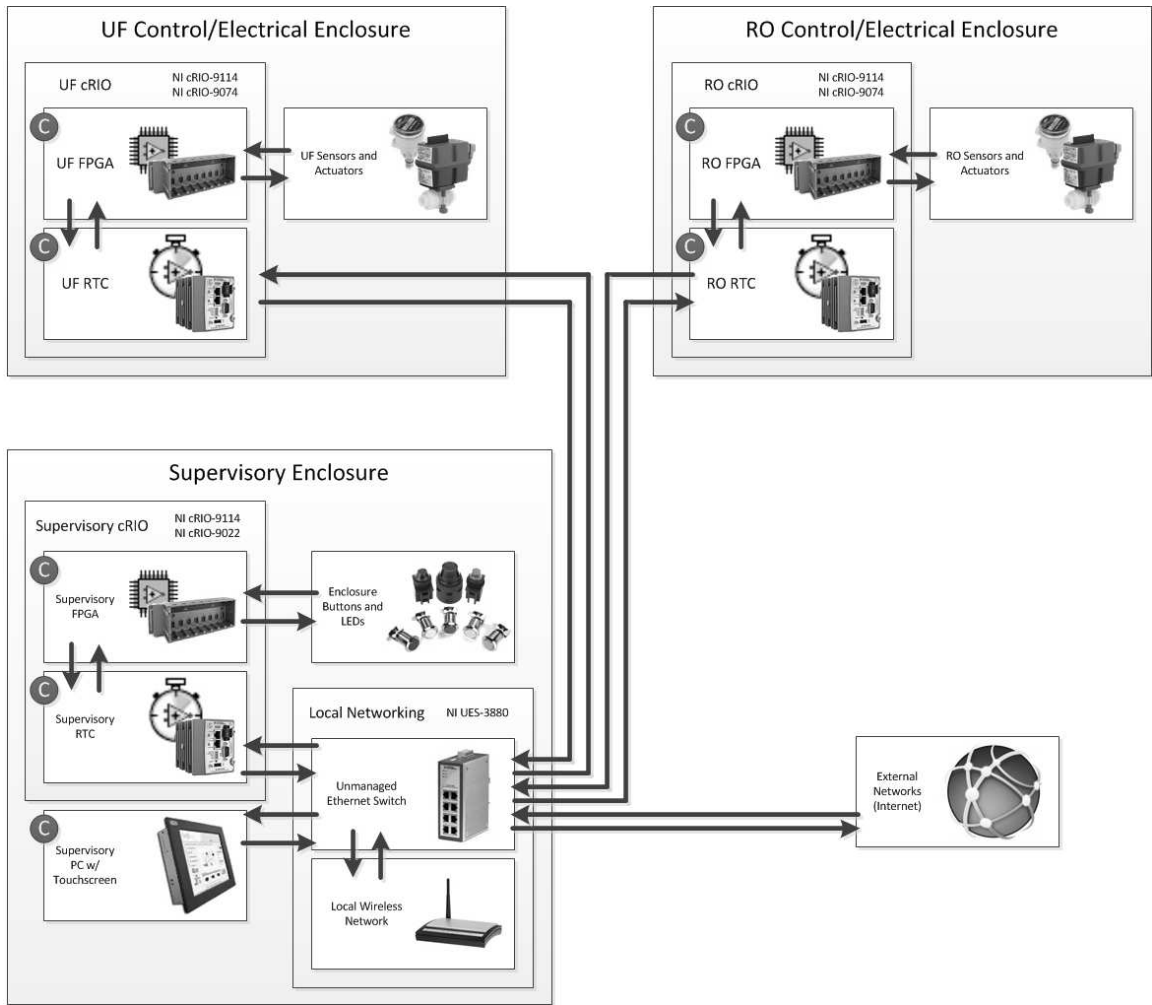
Figure 7.4: CoM2RO distributed control system hardware - physical connections.

7.3.2   Control System Software

UF/RO Field Programmable Gate Array (FPGA) Code

The control system software architecture was designed to allow for the division of tasks and resources among the distributed embedded controllers in the integrated UF/RO system. The lowest level of code on the embedded controllers is the FPGA code, which is responsible for several tasks:

- Collecting data from the analog/digital input modules at the desired sampling rate (for initial operation, sampling is conducted every 200 ms, or at a frequency of 5 Hz).

- Writing analog output data to the analog output modules (for continuous actuator control), and actuating the relay channels on the relay modules (for on/off control of 2-way and 3-way valves, Amiad filter "pause" function, 3-phase contactor power, etc.).

- Simple safety operations (high/low actuator setting cut-offs, system shut-down in the event of connectivity loss, etc.).

- FPGA code initialization/shut-down sequences (turning on/off chassis LED to indicate FPGA code operation, setting I/O channels to defaults).

The FPGA contains the most basic code, but is the most reliable and robust element of the control hardware. This is due to the fact that the circuitry on the

177

FPGA is physically reconfigured when new FPGA code is compiled, resulting in extremely fast and reliable operation. The FPGA code also contains a "front panel" interface for troubleshooting. It is to be noted that the FPGAs on the UF and RO controllers are the same model (see Appendix B for more details regarding controller model/manufacturer), and therefore have the same characteristics of operation.

UF/RO Real-Time Controller (RTC) Code

The RTC utilizes an embedded CPU with dedicated memory to execute the RTC code. The RTC runs in a dedicated operating system (running only the LabVIEW environment), but it can perform the most common LabVIEW functions can be performed on a Windows-based PC. The RTC code can be much more complex and resource-intensive than the FPGA code, but it can be deployed and executed in much less time (FPGA code can take 10-20 minutes to compile and run whereas the RTC code can compile and run in less than a minute).

The RTC code on the UF and RO controllers is responsible for:

- Obtaining and transmitting sensor/control data to/from the FPGA.

- Basic calibrations of sensor data, using either factory calibration values (if the supervisory controller is not present), or field calibration values obtained empirically (when the supervisory controller is connected and operational).

- Storing component locations (i.e., which components are connected to which

channels on the chassis I/O modules) and sorting sensor data so that the sensor measurements appear in the correct indicators on the GUI (also for quick reconfiguration in the event that new equipment is connected, or old equipment wiring is re-arranged).

- Data logging at low frequency (initial frequency of 0.33 Hz) for emergency backup data storage.

- Local control calculations for utilization of basic control algorithms (P/PI/PID), also available in the event that the supervisory controller is not functioning or disconnected.

- Continuous connectivity testing to determine which hardware targets are connected to the local network and are operational.

- Basic process sequences and decision-making logic in the event that the supervisory controller is not functioning or disconnected.

- Remote access for troubleshooting and process monitoring.

The real-time code can also include a process-specific GUI for use in the event that the FPGA/RTC target becomes inaccessible and must be debugged/fixed. This GUI contains a process flow diagram along with displays of all of the local sensor readings and the ability to control the local actuators. Data communication inside (and between) the RTC codes takes place using local and global shared variables.

When the code is deployed to the RTC, variables hosted in the target's shared variable libraries are given space in the local volatile memory. The hardware target keeps track of the location of these memory spaces (sized based on the data type and size of the quantity being stored). When new data are written to the shared variable, this location in memory is overwritten. This is important because it allows the RTC controllers on different frames to operate without errors even if the other hardware targets on the network stop functioning. Connectivity testing (polling the other hardware targets for an active connection) is then used to determine which shared variables to use in making logic/control decisions.

For example, in standard operation (with all 3 RTC codes operational and networked), the supervisory RTC will send sensor calibration values (multiplier and bias) to the UF and RO RTC codes for application to incoming sensor data. In the event that the supervisory RTC code terminates unexpectedly or the hardware becomes disconnected, the UF and RO RTC codes can detect the connectivity loss, and begin to use the factory default calibration values for incoming sensor data. This process can be utilized to switch between the decision making processes located on the supervisory RTC/PC and the emergency sequences located in the individual UF/RO RTC codes. These principles (for the UF and RO RTC/FPGA codes) can be seen in Figures 7.5 and 7.6.
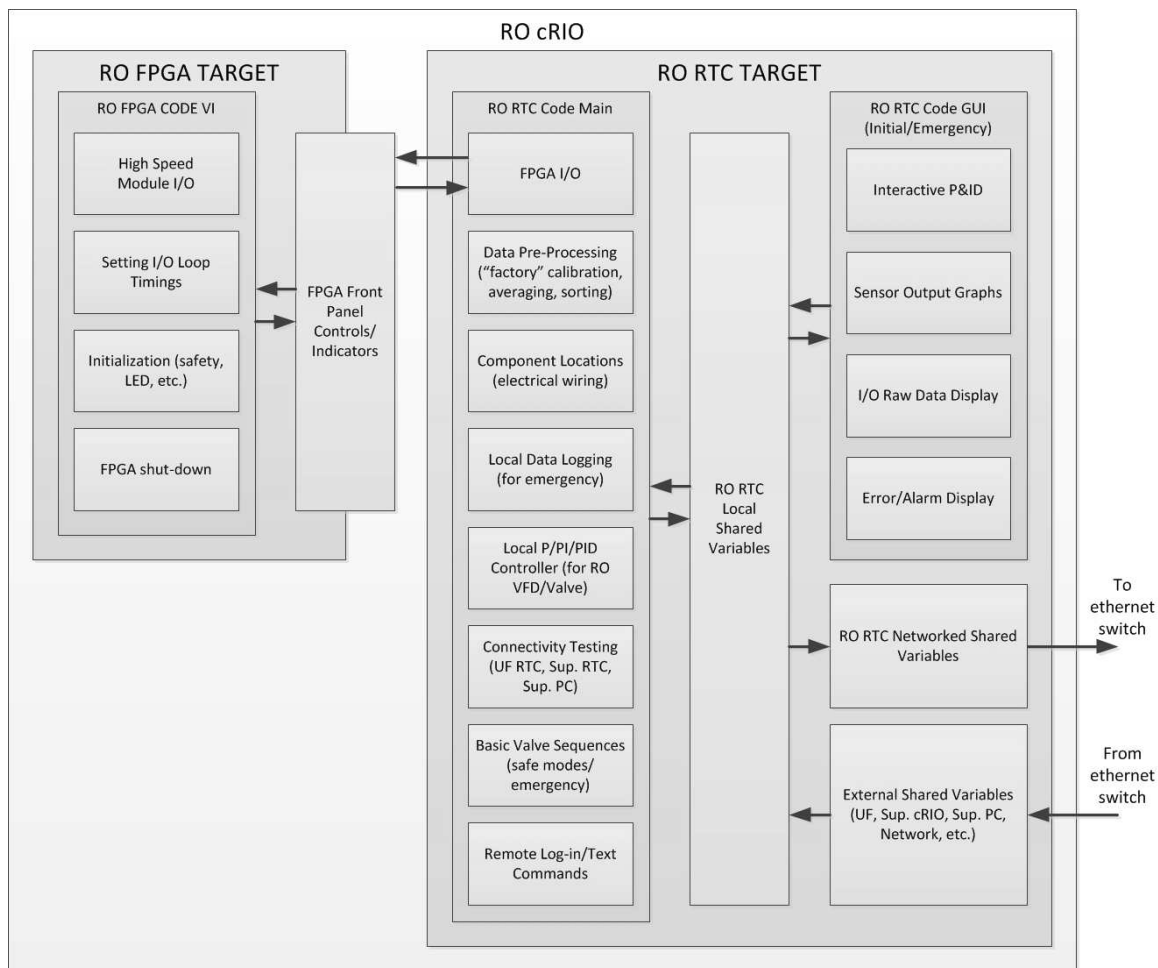
Figure 7.5: CoM2RO distributed control system software - RO system cRIO code architecture.
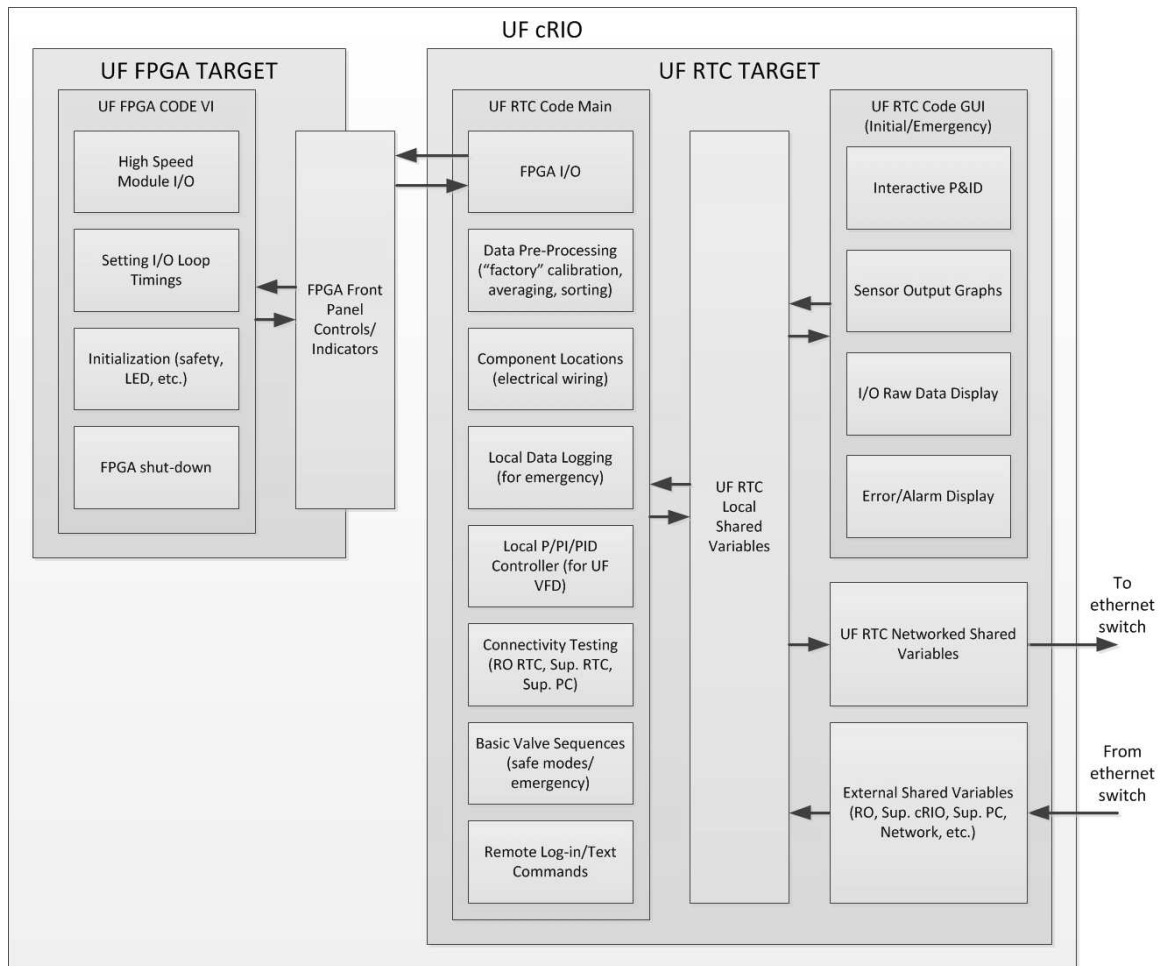
Figure 7.6: CoM2RO distributed control system software - UF system cRIO code architecture.

Supervisory Controller FPGA/RTC Code

The supervisory controller FPGA is used to determine when the supervisory enclosure buttons are pressed (four buttons are present), and also to turn on/off the LEDs (four LEDs are also present on the front of the enclosure). The supervisory RTC code obtains the UF/RO sensor/actuator data by reading the networked shared variables made available in the UF and RO RTC codes. In addition to the aforementioned tasks performed by the local UF/RO RTC codes, the supervisory RTC code is responsible for extensive decision-making logic which is coded in the form of "statecharts". These statecharts are a graphical representation of the decision-making logic code, where operations are represented in the form of states and transitions. The transitions between different flow patterns (backwashing, filtration, cleaning, etc.) are encoded in base-level statecharts that describe the switching of individual valves and the operation of the process controllers. These base-level statecharts are then utilized by higher-level statecharts to sequence the operations in different ways, depending on the process conditions. The higher-level statecharts are coded on the supervisory RTC in order to implement the adaptive backwashing and various other types of process monitoring/control. These statecharts can also be used to implement fault detection, isolation, and fault tolerant control strategies. The supervisory RTC/FPGA code is depicted in Figure 7.7.
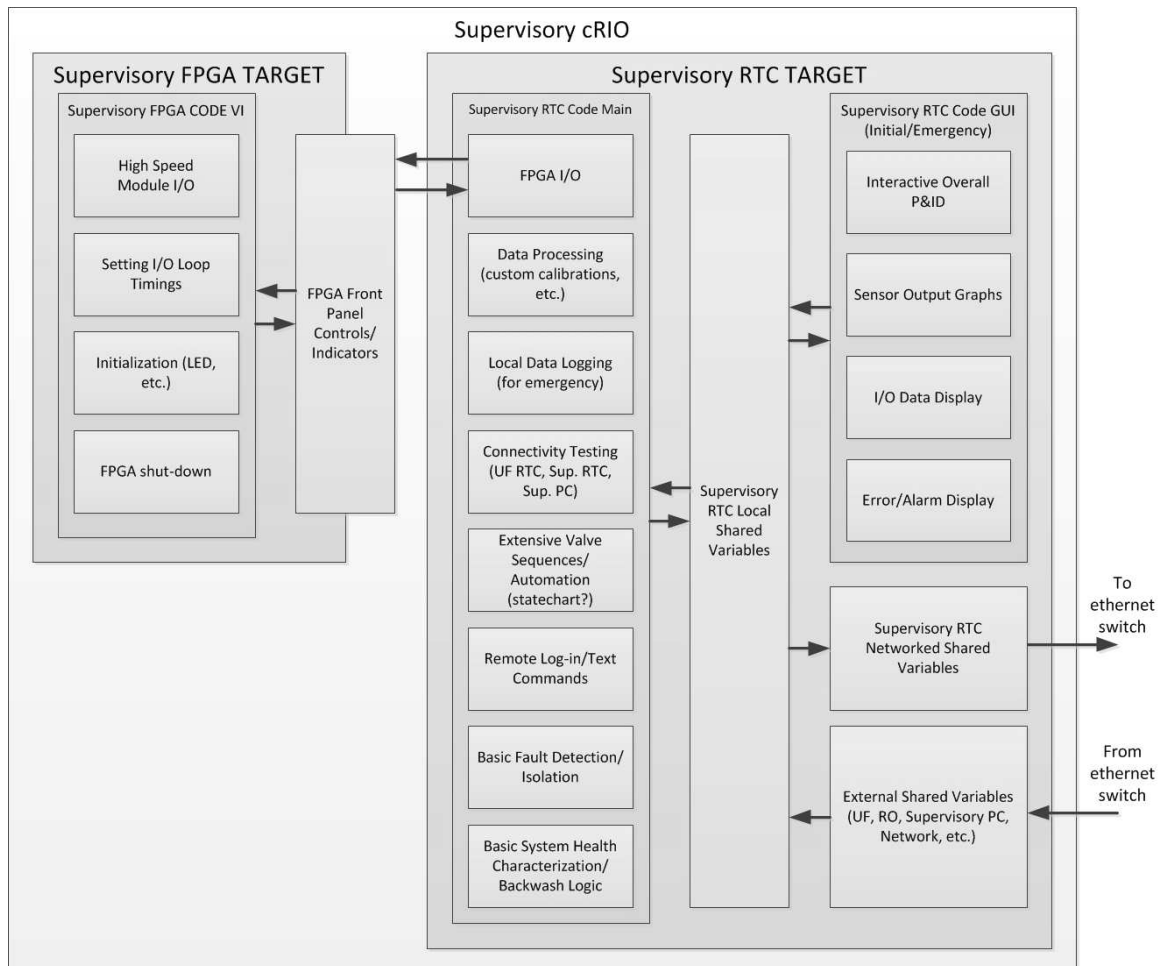
Figure 7.7: CoM2RO distributed control system software - supervisory cRIO code architecture.

Supervisory PC Code

Although the CoM2RO system is designed to operate autonomously with three embedded controllers (FPGA/RTC), basic operator input (such as desired permeate production rate) is necessary. The supervisory PC is also useful for alerting the process operator(s) to faults in the system, as well as to display important system operating parameters. The software on the supervisory PC can include (but is not limited to):

- Displaying an interactive P&ID for easy viewing of current system operation mode.

- Displaying timers for overall system operation as well as time elapsed since last backwash/filtration sequence.

- Calculations of membrane permeability, membrane fouling, trans-membrane pressure, and other important parameters.

- Display of warnings and errors.

- Display of desired production rate.

- Display of results of operating parameter calculations and estimation of UF backwash frequency.

- Connectivity display showing which hardware targets are connected and operational.

- Remote server and user connectivity.

- Interfacing with local/remote codes in a modular fashion for added functionality.

Similar to the supervisory RTC, the supervisory PC can access the networked shared variables (when operating the LabVIEW-based GUI). Alternatively, the supervisory RTC sends the desired data over the network using the UDP network protocol. This allows programs written in other languages (MATLAB, C, etc.) to utilize the supervisory data stream and interact with the system (two-way communication; the external software can send data to the supervisory RTC as well as receive data). These external programs (or remote web-based applications) can utilize the additional resources on the supervisory PC to conduct calculations or decision-making logic, while communicating with the supervisory RTC in a modular fashion. The supervisory PC code architecture is shown in Figure 7.8.
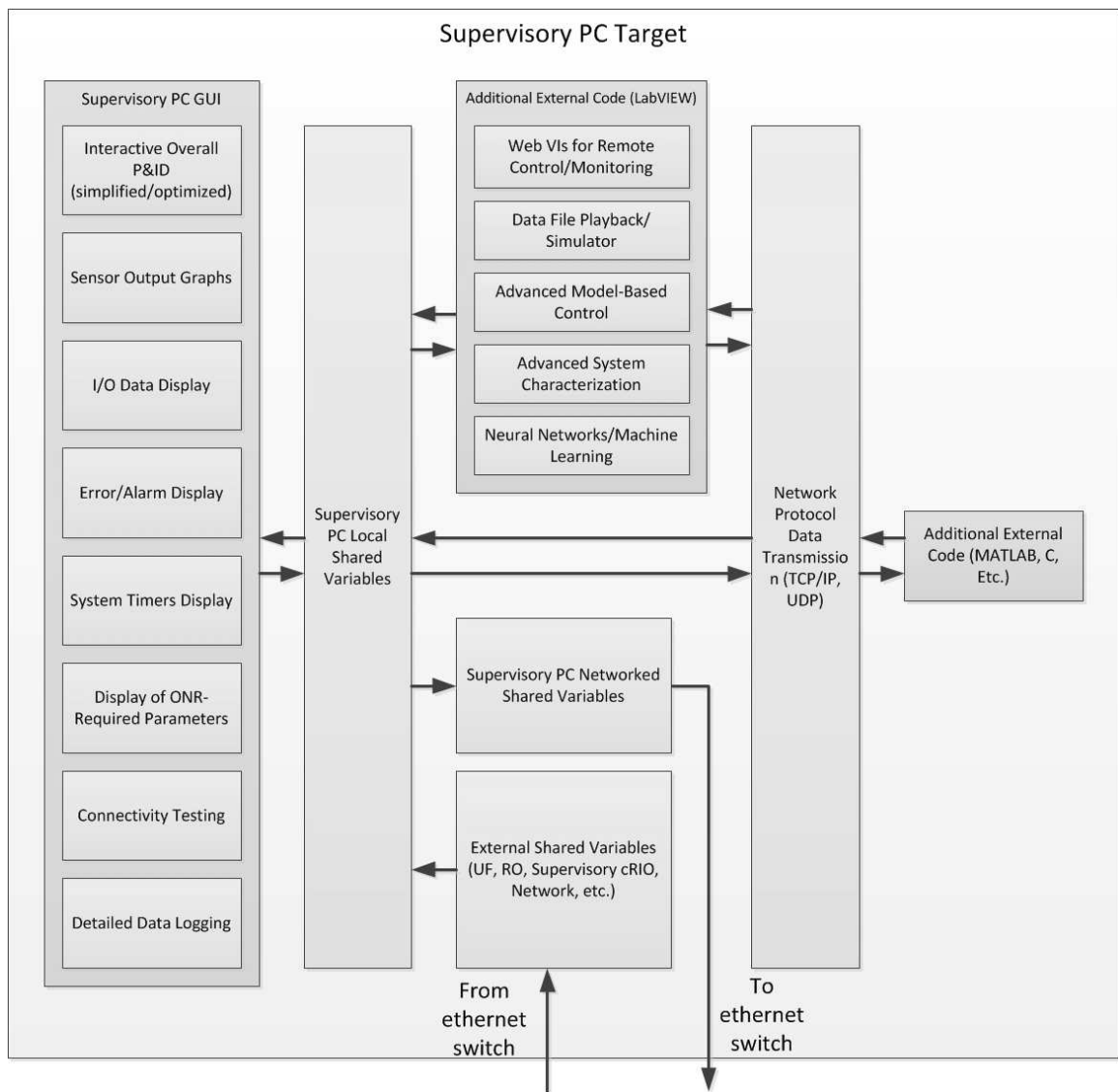
Figure 7.8: CoM2RO distributed control system software - supervisory PC code architecture.

7.4   System Control and Optimization Algorithms

7.4.1   Basic Controller Set-up

The CoM2RO system software relies on multiple control loops to ensure operation at the correct process conditions (e.g., feed flow rate, feed pressure, permeate production rate). In the initial stages of design and testing, three control loops were used (see Fig. 7.9). System actuators include the UF VFD on the UF centrifugal feed pump (controls pressure/flow in the UF system), the RO VFD on the positive displacement RO feed pump (controlling feed flow rate to the RO system and also influencing pressure), and the actuated retentate valve (controlling the RO system pressure).

These actuators can use any of the sensor measurements as the process variable when the embedded control systems (UF/RO/supervisory) are connected across the local network. However, in order to ensure the safe and reliable system operation, the most important process variables have been identified, and the control system was designed around these sensor readings (e.g., feed flow rate, feed pressure, pressure at the RO pump inlet, permeate flow rate). As described in the operability subsection (Sec. 7.1.2), the UF system operates according to the demand of the RO system (up the maximum flow rate and pressure allowed by the UF system; 60 gpm and 70 psi, respectively); keeping the RO feed pump inlet pressure at the set-point while allowing the RO pump to deliver the desired feed flow rate.
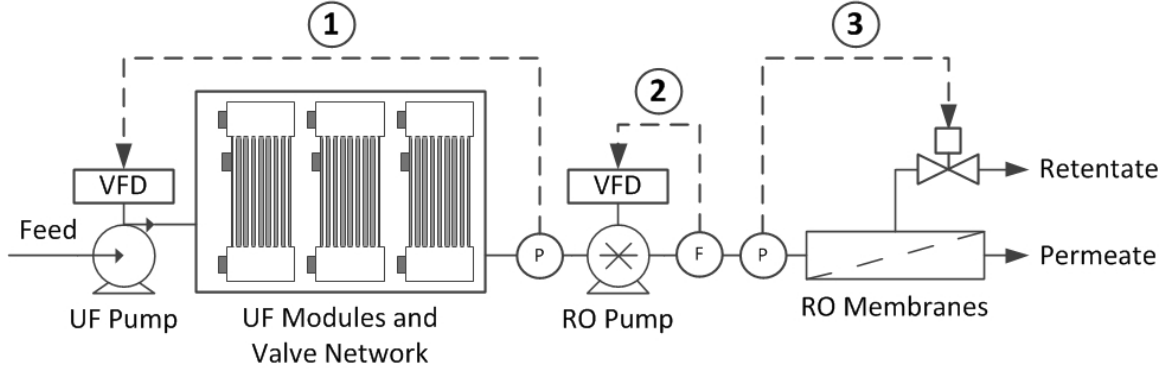
Figure 7.9: CoM2RO system basic control loop structure.

The basic control loop structure depicted in Fig. 7.9 has the following features and functions:

- Control loop 1 uses the inlet pressure of the high-pressure positive displacement RO pump as the measured variable. The goal of this loop is to adjust the UF feed flow rate and pressure so that the RO feed pump can deliver the desired amount of feed (and ultimately, product) water. This loop also ensures the safety of the RO pump operation and prevents cavitation due to situations where the inlet pressure may be lower than the required net positive suction head ($NPSH_r$, the minimum pressure required at the pump inlet). A PI controller of the following form is used:

$$VFD_{set}^{UF} = K_p^1(P_{RO\ inlet}^{sp} - P_{RO\ inlet}(t)) + \frac{K_p^1}{\tau_i^1} \int_0^t P_{RO\ inlet}^{sp} - P_{RO\ inlet}(\tau))d\tau \quad (7.1)$$

189

where VFD_{set}^{UF} is the speed applied to the UF pump motor (in RPM), K_p^1 is the proportional gain (in RPM/psi), $P_{RO\,inlet}^{sp}$ is the RO pump inlet pressure set-point (in psi), $P_{RO\,inlet}(t)$ is the current measured RO pump inlet pressure (in psi), and τ_i is the integral time constant (in seconds).

For initial testing of the RO pump inlet pressure controller, $K_p^1 = 5.5\,RPM/psi$ and $\tau_i^1 = 0.05\,s$. These controller parameters were determined empirically through testing of a range of proportional gains and integral time constants (these specific values were chosen to minimize controller overshoot and convergence time to the pressure set-point). A maximum rate of change of the control action with time can also be used; this value can be set by the user on the GUI front panel for fine-tuned control over the system dynamics.

- Control loop 2 uses the feed flow rate to the RO membranes to adjust the RO pump VFD setting. For initial testing of the RO VFD controller, a linear relationship between the motor speed (RPM) and feed flow rate (in GPM) was used:

$$VFD_{set}^{RO} = (43.103\,\frac{RPM}{GPM})Q_f^{desired} - 29.009\,RPM \qquad (7.2)$$

where VFD_{set}^{RO} is the speed applied to the RO pump motor (in RPM) and $Q_f^{desired}$ is the desired RO process feed flow rate (in GPM).

190

The above empirically determined relation is very close to the manufacturer suggested relation (proportionality constant of 40 RPM/GPM and an offset of zero) and can be used as a "model" to set the RO VFD to the appropriate speed. A PI controller is also present in the system code and can be activated instead of this simple relation if desired.

- Control loop 3 uses the RO feed pressure to control the actuated retentate valve position. This valve controller can also use measurements of permeate flow rate, RO membrane module feed pressure, or various other process variables (e.g., RO retentate pressure, UF backwash pressure) as feedback. The following PI controller is used to maintain the RO module feed pressure at the set-point:

$$Valve_{set} = K_p^2(P_{RO\,feed}^{sp} - P_{RO\,feed}(t)) + \frac{K_p^2}{\tau_i^2}\int_0^t P_{RO\,feed}^{sp} - P_{RO\,feed}(\tau))d\tau \quad (7.3)$$

where $Valve_{set}$ is the position of the actuated retentate valve (in % open), K_p^2 is the proportional gain (in %/psi), $P_{RO\,feed}^{sp}$ is the set-point feed pressure to the RO membrane modules (in psi), $P_{RO\,feed}(t)$ is the current feed pressure to the RO membrane modules (in psi), and τ_i^2 is the integral time constant (in seconds).

For initial testing, $K_p^2 = -0.001\ \%/psi$ and $\tau_i^2 = 0.01\ s$. These controller parameters were determined empirically through testing of a range of proportional gains and integral time constants (these specific values were chosen to minimize

controller overshoot and convergence time to the pressure set-point). This loop also can be operated with a proportional controller and a maximum rate of change of the valve position with time (user-defined from the GUI front panel). Gain scheduling (using different gain and time constant values for different ranges of $|(P^{sp}_{RO\,feed} - P_{RO\,feed}(t))|)$ was also employed.

7.5 Preliminary CoM2RO Experimental Results

The CoM2RO system was tested at the UCLA co-generation plant with feed water from the cooling tower blow-down stream (discarded water from the cooling cycle with relatively high levels of mineral scalants and anti-scalant chemicals, see Table 7.1). The water quality and selected properties are shown below in Table 7.1. There is a wide range of water quality due to the seasonal/daily variations in temperature, evaporation rate, make-up water quality, feed water quality, and proprietary additives used by the power plant operators.

In the following experiments, the UF VFD controller (presented in Eq. 7.1) was tested in the case where the UF flow mode is switched from normal filtration flow (all three UF modules in filtration mode), to a mode where one of the modules is backwashed while the other two are in normal filtration. Without the controller, the pressure at the inlet to the RO pump fell to below 14 psi (the pump can be damaged if the inlet pressure falls below this threshold), and can even decline to a level that

Table 7.1: Co-generation plant water quality.

Conductivity	$2300 - 3500$	μS
Estimated TDS	$1000 - 2000$	mg/L
pH	$7 - 8$	
Turbidity	$1.34 - 2.43$	NTU
Calcium Hardness	< 600	ppm
Chlorine	$< 0.5 - 1.5$	ppm
Temperature	$70 - 80$	$^{\circ}F$

could (below 5 psi) open the circuit in the low pressure switch connected to the RO pump inlet line, powering down the RO VFD and halting system operation. The objective of this controller was to maintain the inlet pressure to the RO pump above the 14 psi threshold while the system is switched between filtration and backwashing modes. The second goal of the experiments was to test the RO retentate valve controller (presented in Eq. 7.3) for set-point transitions in the feed pressure to the RO membrane modules (instigated by the user or by the automated control algorithms) and also for disurbance rejection capabilities (e.g., when the RO feed pump speed is changed).

The presented experiments were conducted at a UF system feed flow rate of 31 gpm, with an RO feed pressure of 311 psi. In order to prevent particles from fouling the RO membranes, the UF system was dosed with a coagulant (ferric chloride, $FeCl_3$,

approx. 10 ppm). It was also necessary to protect the RO membranes from the chlorine present in the cooling tower water, and this was accomplished by adding sodium metabisulfite ($Na_2S_2O_5$, 2 ppm) during system operation. Both additives were injected into the process using metering pumps (controlled by user through the system GUI software).

The controllers detailed in Sec. 7.4 were implemented and used to set the system operating state (UF feed flow rate/pressure, RO feed flow rate/pressure). During the automated sequence operation, it was required to keep the RO pump inlet pressure above 14 psi (cavitation and pump damage can occur at inlet pressures below 14 psi) while the pre-filtration system switched one UF module from normal filtration to backwashing mode. In the experimental results presented in Fig. 7.10, the controller on the UF VFD was tasked with maintaining the pressure at a set-point of 25 psi while the system transitioned each module from filtration to backwash, and then back to filtration mode.

The data in Fig. 7.10 show the RO pump inlet pressure and UF pump speed during the transition of each of the three UF modules from backwash to filtration mode. It can be seen that the UF pump speed controller was able to maintain the RO pump inlet pressure above the minimum value of 14 psi (minimum values seen in experiments are 16-17 psi) and return the RO pump inlet pressure to the set-point of 25 psi within approximately 5 seconds. Without the controller, the system pressure would drop below the cavitation threshold for the RO pump, and may decline to a
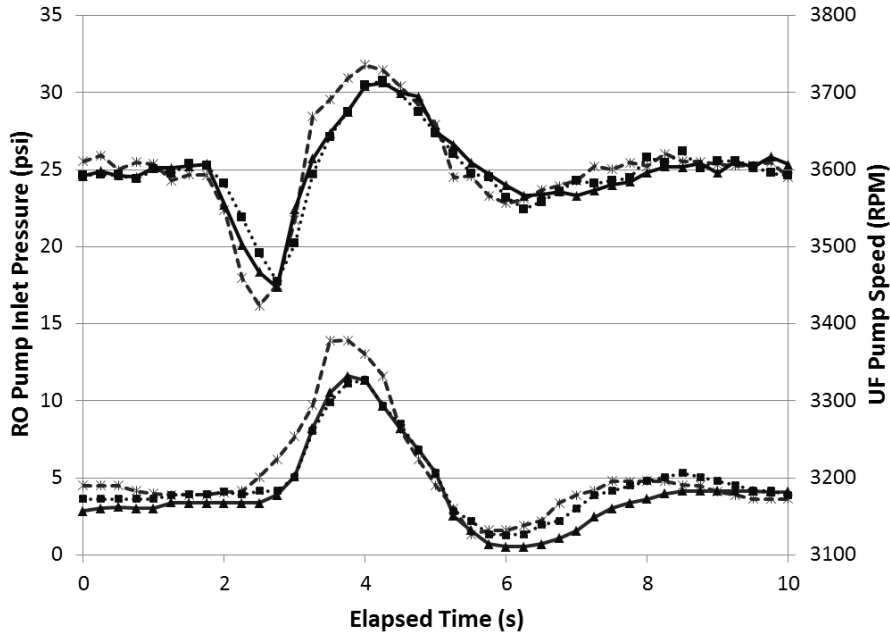
Figure 7.10: Control of RO pump inlet pressure during transition from backwash to filtration mode showing RO pump inlet pressure and UF pump motor speed for three transitions (one full backwash cycle, one transition for each UF module). The upper curves represent the RO pump inlet pressure, while the lower curves represent the UF pump speed. The solid lines (with triangle markers) denote the transition of UF-101, the dashed lines (with "x" markers) denote the transition of UF-102, and the dotted lines (with square markers) denote the transition of UF-103. The RO pump inlet pressure set-point was set to 25 psi, while the proportional gain of the proportional controller used on the UF pump was set to $K_p = 5.5 \, RPM/psi$. The system feed flow rate was set at 31 gpm, RO feed pressure was maintained at 311 psi, the feed conductivity was approximately 2600 μS, the RO system recovery was 52%, and the water temperature was 75 $^\circ F$.

level that could open the circuit in the low pressure switch (below 5 psi). In this case, the system would cut power to the RO VFD, and would completely shut down.

More advanced control experiments were conducted utilizing a pulse backwash (in addition to standard retentate backwash) of the UF modules. In the pulse backwash, the system's accumulators (two accumulators with a volume of 5 L each) are charged up to approximately 41 psi, and then the UF module feed channel is opened to the drain. This drop in feed-side resistance causes the discharge of the accumulators, and in turn, a high-flux (approximately 205 L/m^2h) pulse of water travels through the UF membrane. For the experimental results shown in Fig. 7.11, three pulses were used to dislodge foulants on the feed side of the UF membrane. During the pulse backwash, the UF pump speed controller was able to keep the RO pump feed pressure at the set-point of 25 psi (with a standard deviation of 0.58 psi during the time span shown in the figure). These data demonstrate that the high-level automation controlling the individual valve sequences (and in turn, the flow modes of the system) can successfully work in tandem with the process control algorithm for the UF pump speed.

Data in Figs. 7.12 and 7.13 demonstrate the use of a proportional controller (see Eq. 7.3) with gain scheduling for control of the actuated retentate valve. The actuated retentate valve is used to control the feed pressure entering the RO membranes to the desired set-point, which is necessary in situations where the speed of the RO pump is changed by the control system. A proportional controller with gain scheduling was found to be effective for set-point transitions in feed pressure, and the gain scheduling
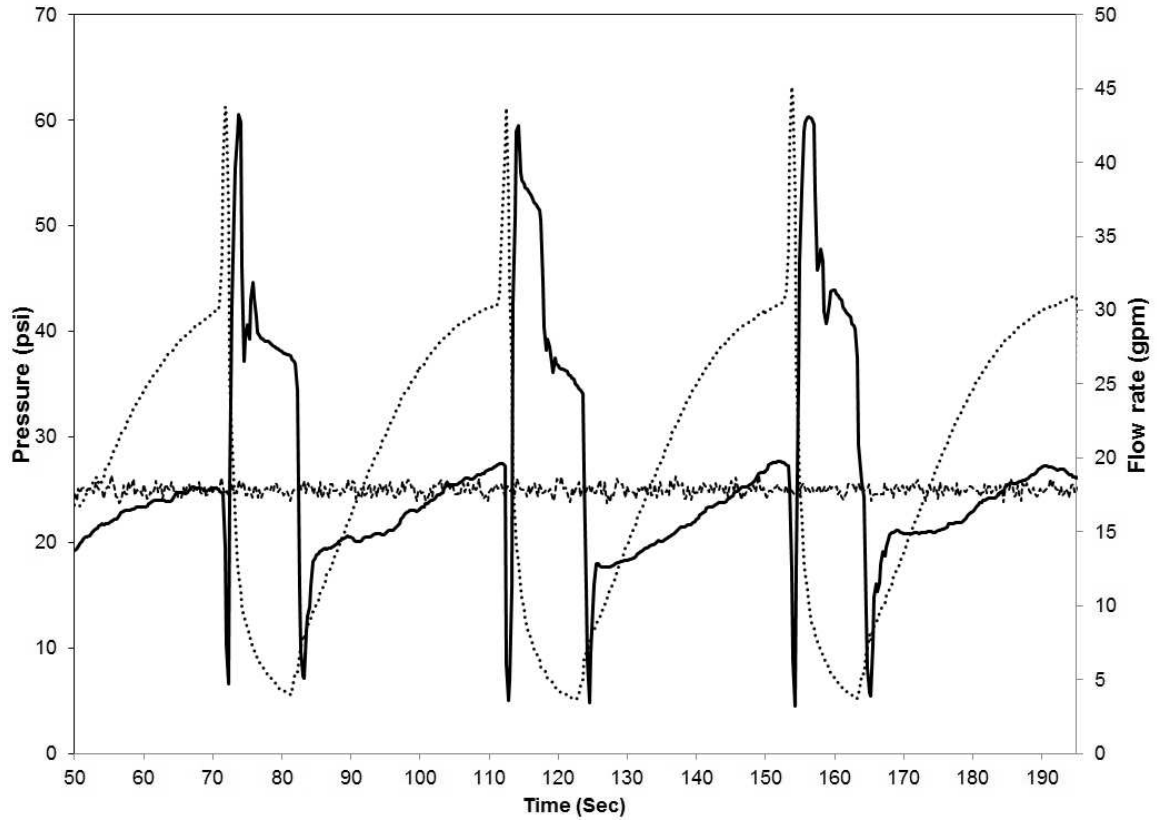
Figure 7.11: Use of automated control system with control of RO pump inlet pressure showing system pulse backwashing cycle using accumulators. The solid line denotes the backwash flow rate into the module, the dotted line denotes the backwash line pressure, and the dashed line represents the RO pump inlet pressure during the pulse backwashing process (individual data points are replaced by lines for clarity). The RO pump inlet pressure set-point was set to 25 psi, while the proportional gain of the proportional controller used on the UF pump was set to $K_p = 5.5\,RPM/psi$. The system feed flow rate was set at 30 gpm, RO feed pressure was maintained at 320 psi, the feed conductivity was approximately 3500 μS, the RO system recovery was 38%, and the water temperature was 75 $^\circ F$.

parameters are listed in Table 7.2.

Table 7.2: RO valve controller gain scheduling parameters.

Pressure Error Range	Proportional Gain	
$|(P^{sp}_{RO\,feed} - P_{RO\,feed}(t))| < 15\,psi$	$K_p = -0.001$	%/psi
$|(P^{sp}_{RO\,feed} - P_{RO\,feed}(t))| > 15\,psi$	$K_p = -0.01$	%/psi
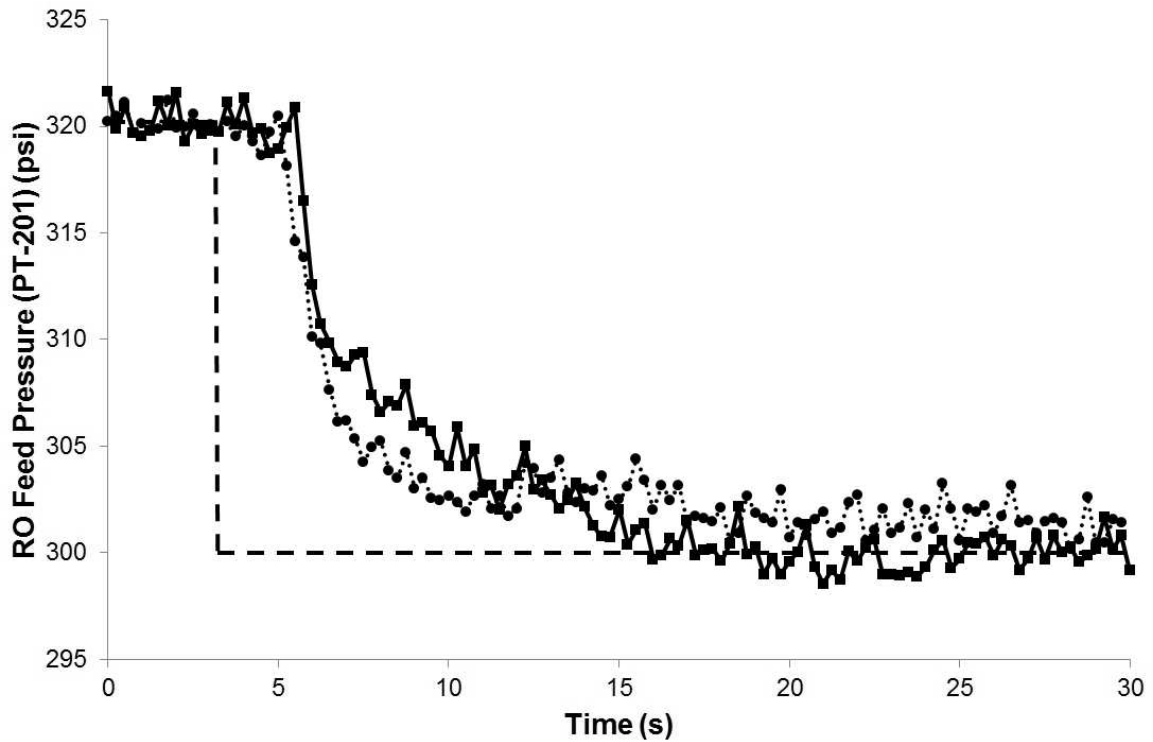
Figure 7.12: RO feed pressure during two set-point transitions from 320 psi to 300 psi using a proportional controller with gain scheduling on the actuated retentate valve. The data points are denoted by markers (squares for the first transition, and circles for the second transition) with connecting lines used for clarity (a solid line for the first transition data set, and a dotted line for the second transition data set). The controller used two ranges of proportional gain: $K_p = -0.001\ \%/psi$ for $|(P^{sp}_{RO\ feed} - P_{RO\ feed}(t))| < 15psi$ and $K_p = -0.01\%/psi$ for $|(P^{sp}_{RO\ feed} - P_{RO\ feed}(t))| > 15\ psi$. The system feed flow rate was set at 27 gpm, the feed conductivity was approximately 3000 μS, the RO system recovery was 50%, and the water temperature was 76 $^\circ F$.
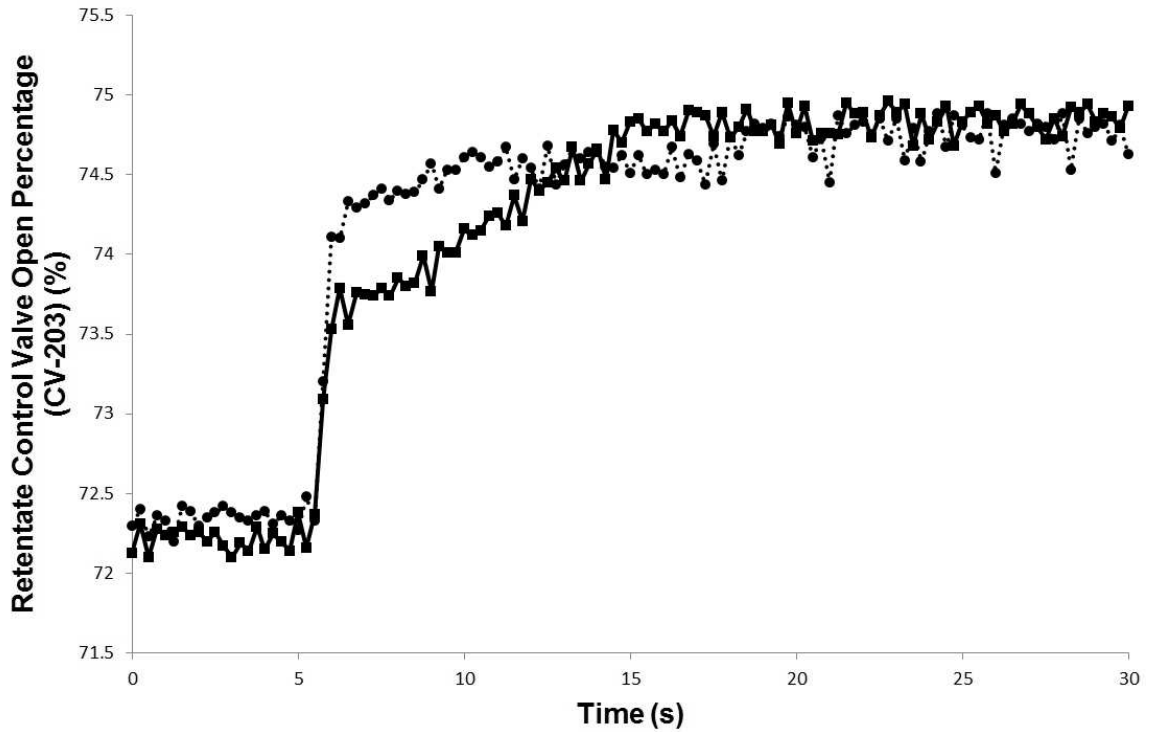
Figure 7.13: Actuated retentate valve position (control action) during two RO feed pressure set-point transitions from 320 psi to 300 psi. The data points are denoted by markers (squares for the first transition, and circles for the second transition) with connecting lines used for clarity (a solid line for the first transition data set, and a dotted line for the second transition data set). The controller used two ranges of proportional gain: $K_p = -0.001 \%/psi$ for $|(P^{sp}_{RO\,feed} - P_{RO\,feed}(t))| < 15\,psi$ and $K_p = -0.01 \%/psi$ for $|(P^{sp}_{RO\,feed} - P_{RO\,feed}(t))| > 15\,psi$. The system feed flow rate was set at 27 gpm, the feed conductivity was approximately 3000 μS, the RO system recovery was 50%, and the water temperature was 76 $^\circ F$.

7.6 Future Work

Future work with control and optimization of the CoM2RO system can include (but is not limited to) implementation of model-based controllers for VFD/actuated valve control (similar to Chapters 4 and Appendix A) in order to further minimize system energy usage and to optimize set-point transitions in the presence of feed quality disturbances. Control algorithms to conduct fault detection, isolation, and fault-tolerant control can also be implemented in order to prevent system shut-down/damage in the event of an actuator fault, or to alert the user about a damaged component before system shut-down is necessary. It is also possible to augment the CoM2RO with the MeMo detector (as described in Chapter 6) and MSIA software for use with feed water of high scaling propensity.

Multiple studies can be carried out with a range of source waters (seawater, agricultural drainage water, municipal wastewater, etc.) to determine the optimal backwash strategies and control logic to allow for low-maintenance, extended system operation. It is also noted that the CoM2RO system can be used as a modular platform to rapidly evaluate novel components (RO membranes, filters, pumps, energy recovery devices, etc.) that may be developed in the future.

Chapter 8

Conclusions

The objective of this dissertation was to develop advanced ("smart") methodologies for reverse osmosis (RO) desalination system control, optimization, operation and monitoring, and to evaluate their performance on experimental commercial-scale RO processes. The development of advanced control algorithms for RO desalination systems is critical to addressing the need for the future development of fresh water supplies, as well as to upgrade existing systems to allow for higher yield and more cost-effective operation. Addressing these issues, the results presented in this work demonstrate the viability of several types of RO system control and optimization, both for overall system control and for specific applications such as membrane cleaning/mineral salt scaling prevention. The dissertation also provides a background on the concepts, design, construction, operation, and testing of the experimental water purification systems built at UCLA.

A first-generation RO water desalination system (M3) was designed and constructed as a test platform for the control and monitoring methods presented in this dissertation. A fundamental dynamic model was developed to describe the M3

202

reverse osmosis water desalination system (Chapter 3) in order to develop model-based control algorithms for RO system control; the parameters of this model were then computed using step test data from the M3 system. Specifically, correlations were derived in order to relate the actuator position to model parameters (creating a mathematical relationship between the model outputs such as valve resistance and the physical actuator settings), and the remaining model parameters were computed based on the experimental data.

Since the success of RO system operation is highly dependent on incoming feed water quality, it was desired to design and test model-based controllers that could maintain system operation at user-specified set-points (system pressure, flow rates, recovery) in the presence of feed water quality disturbances. In Chapter 4, a nonlinear model-based control strategy was developed and experimentally implemented on the experimental M3 RO system for the purposes of efficiently conducting set-point transitions and rejecting disturbances (e.g., large changes in feed water quality). This nonlinear controller was implemented to manipulate the retentate stream actuated valve along with a proportional-integral controller employed to manipulate the variable frequency drive speed adjusting the feed flow rate. Performance of the nonlinear controller was compared to the performances of proportional and proportional-integral control algorithms, as well as benchmarked against a simulated nonlinear model-based controller during retentate flow rate set-point transitions. It was demonstrated that the nonlinear controller is much better suited to deal with the highly coupled closed-

loop system dynamics during set-point transitions (the nonlinear controller reached the set-point in approximately a minute, whereas the P and PI controllers reached the set-point in 5-10 minutes while displaying large oscillations in flow rate and system pressure) and was shown to outperform the traditional control schemes. The model-based nonlinear controller was also able to maintain system operation at the user-specified set-points when the experimental reverse osmosis system was subjected to a series of large step changes in feed salt concentration.

Once the controller was designed in order to conduct efficient set-point transitions and reject disturbances in feed water quality, it was desired to optimize the steady-state operation of the M3 RO system with respect to system energy consumption while maintaining a user-specified permeate flow rate. Additionally, it was desired to use a model-based optimization algorithm to calculate the appropriate system set-points (feed flow rate and system recovery via retentate valve position) for varying feed water quality.

In Chapter 5, an optimization-based control strategy was developed and experimentally implemented on the M3 reverse osmosis membrane desalination system. First, the system model derived in Chapter 3 was combined with a model for RO system specific energy consumption to form the basis for the design of an optimization-based control system. The control system uses real-time sensor data and user-defined permeate flow requirements to compute in real-time the energy-optimal set-points for the retentate valve position and feed flow rate. Implementation of the control

system on UCLA's M3 RO system demonstrated its ability to achieve energy-optimal operation that is very close (observed errors of 9-20% between experimental data and theoretical SEC curve at low system recovery values, and observed errors of 1-13% for the higher, energy optimal system recovery values) to the theoretically predicted energy consumption curves.

Although the previously developed controllers use the system model to efficiently control the steady-state and transient operation of the overall RO system (system pressure, flow rates, recovery), they do not account for scaling/fouling which may occur on the RO membrane surface. For systems treating feed water with high mineral scaling propensity, it was desired to develop a method for detecting mineral salt scaling on the RO membrane surface and initiating RO membrane scale mitigation in an automated fashion.

For the purposes of scale detection, an approach to real-time monitoring and analysis of the formation of mineral scale on reverse osmosis (RO) membranes was developed using an ex-situ direct observation membrane monitor (MeMo) (Chapter 6). Real-time images of the membrane surface in the MeMo membrane channel were analyzed online to detect the onset of mineral crystals and to monitor the evolution of the fractional coverage by mineral salt crystals and crystal count. Membrane surface image analysis (MSIA) software, which was developed specifically for the MeMo system, was capable of real-time detection of the formation and growth of mineral crystals on the membrane surface. The automated image analysis program

(operating either on-line, or in a post-processing mode) was shown to accurately determine the membrane surface coverage by mineral salt scaling and the number of crystals present in the observation area of the detector.

It was also demonstrated, for the first time, that the MeMo detector and MSIA software could be coupled with automated feed flow reversal (FFR) triggering in order to mitigate mineral salt scale formation on the RO membrane surface. The effectiveness of this monitoring and control approach was demonstrated on two pilot systems; a brackish water RO (BWRO) system operating at Ben-Gurion University in Israel (for the FFR triggering experiments showing the proof-of-concept) and the M3 RO desalination system at UCLA (demonstrating the expanded use of the MeMo detector and MSIA software for multi-cycle FFR scale mitigation at high recovery). Multi-cycle FFR operation using automated FFR initiation also proved that this strategy can be used to recover permeate flux lost due to mineral salt scaling and extend the operating life of RO systems under scaling conditions.

Based on the concepts developed in Chapters 3 - 6 and experience gained with the experimental M3 system operation in the lab and in the field, a second-generation UF/RO system was designed and constructed with the goal of improving system operation. The design and construction of the CoM2RO system (detailed in Chapter 7) allows for integrated MF/UF/RO system operation with optional usage of an intermediate tank. The CoM2RO system is a modular system with a small footprint, containing a distributed control system designed to employ adaptive backwashing,

fault-detection, and model/optimization-based control. These novel control strategies are utilized in order to allow for operation of the CoM2RO system with almost any type of source water, for operation utilizing energy optimization and automatic cleaning, the ability to conduct fault handling, and to assist the user in any necessary decision-making processes.

Basic system control loop operation is detailed, demonstrating the effectiveness of the implemented P/PI control algorithms in controlling the UF pump motor speed (through the VFD) in order to maintain the RO pump inlet pressure above the low-pressure threshold of 14 psi during transitions in system modes (e.g., the transition from normal filtration to retentate backwash, or in the case of pulse backwashing with RO retentate). It was also shown that operating the actuated retentate valve according to the P/PI control algorithm using gain scheduling was effective in maintaining the feed pressure to the RO membrane modules at the specified set-point (within 2 psi). It was shown that the system can successfully treat the power plant cooling tower water through automated backwashing sequences (e.g., retentate backwash, pulse backwash) while the controllers maintain the system operating conditions (feed flow rate, RO pump inlet pressure, RO feed pressure) at the desired set-points, demonstrating the viability of the software architecture for fully automated system operation.

In summary, this thesis details the development of novel methods of UF/RO control and monitoring for reverse osmosis water desalination and implementation on

laboratory and field pilot systems in order to improve/optimize UF/RO system operation. Control methodologies for steady-state, transient, and disturbance rejection behavior of RO systems were developed through first-principles models and applied to a first-generation experimental system which was designed and constructed at UCLA. Novel monitoring methods were also developed for detection of the onset of mineral scaling, and for the first time, were used in an automated fashion to trigger RO membrane cleaning. The effectiveness of the monitoring and control algorithms detailed in this thesis was evaluated, and shown to greatly improve traditional RO system operation. A second-generation RO desalination system was subsequently designed and constructed in order to improve upon the operability, versatility, and modularity of the first-generation system. This second-generation system has the capabilities to utilize all of the control methods developed for the first-generation M3 RO system, in addition to the integrated system operation utilizing the UF pre-filtration process, and provides a unique platform for advancing the science of water purification and RO desalination.

A   Model Predictive Control of Feed Flow Reversal Actuation for Water Hammer Avoidance

Building on the feed flow reversal process detailed in Chapter 6.6, a special case was analyzed for an RO system that is not outfitted with a variable frequency drive (VFD) on the feed pump. This specialized case utilizes an additional actuated valve (bypass valve) in order to control the feed flow rate entering the FFR valve manifolds (and subsequently, the RO membranes). Although such an approach is not a practical long-term solution for an RO plant, it could be useful for interm plant retrofitting or for emergency operation in the event of a VFD failure.

Overview

In reverse osmosis processes, particularly with brackish water feeds or processes running at a high level of recovery, dissolved ions can precipitate out of solution and crystallize in the bulk or directly on the membrane surface leading to mineral scaling. Scale formation on the membrane surface leads to decreased permeate productivity [41], and can result in permanent membrane damage if scaling is allowed to progress significantly past its initial stages. Several methods are currently used to mitigate scale formation in RO systems; addition of anti-scalant chemicals to the feed [86], flushing the membrane units with low-TDS (total dissolved solids) permeate water

[39], or chemical cleaning of the RO membranes. These current methods of scale mitigation have several disadvantages. Anti-scalants are only useful to a degree, and if added in excess, could promote biofouling [75]. The cost of the anti-scalants is also an important consideration [23]. In the case of the permeate flush, this process will require halting the reverse osmosis operation (i.e., cessation of permeate production) for a sufficiently long period to allow for system flushing and may even using up some of the previously produced permeate water. A recent method of RO operation called feed flow reversal has been developed, that prevents scale formation without the addition of expensive chemicals or extensive periods of system down-time [71]. In this approach, a system of solenoid valves are used around the membrane modules configured specifically so that the direction of the feed flow through the membrane units can be reversed (see Fig. A.1). This reversal of the feed flow also reverses the axial salt concentration profile [40] at the surface of the membrane, thereby exposing the mineral salt scaled areas to an undersaturated feed (w.r.t. scalants) leading to dissolution of the surface mineral crystallization [71].

It is imperative to operate the flow reversal process for the optimal length of time; switching back to normal flow too quickly may leave scale crystals on the membrane, while operating the flow reversal for too long may cause scale to form on the outlet end of the membrane surface.

When operating a system that utilizes feed flow reversal, it is important to carry out the mode switching (from forward flow to reverse flow) in a manner which will
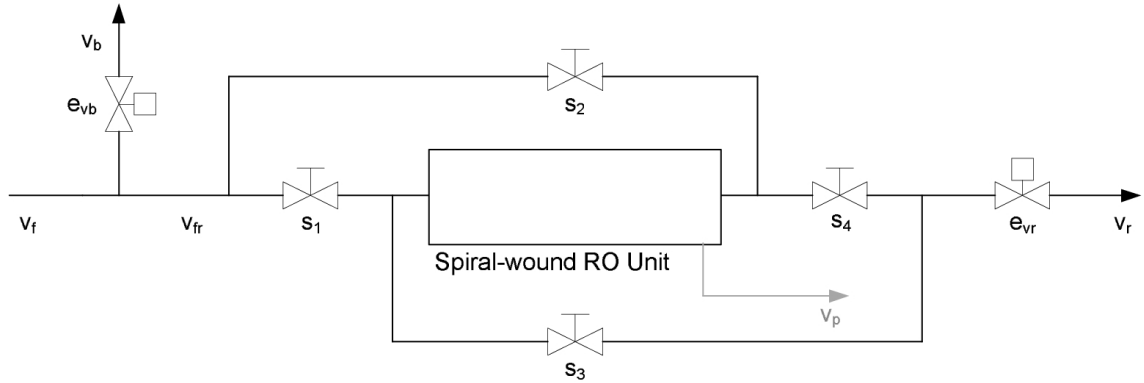
Figure A.1: An expanded view of the flow reversal configuration surrounding the spiral-wound unit

not cause water hammer. The phenomenon of water hammer takes place when fluid moving at a moderate to high velocity ($> 1.5\frac{m}{s}$ for the example discussed below) suddenly encounters a blockage in the pipe (for instance, when a solenoid valve is closed). The fluid's inertia causes a step change in the velocity (i.e., to zero velocity), causing a pressure wave which can damage the process equipment [53]. The Joukowski formula [70] can be used to estimate the magnitude of the pressure wave caused by water hammer for a given system configuration, and it is seen that this problem can be especially prominent in large systems with a high feed flow rate.

For example, consider the system presented in Fig. A.3. Initially, solenoid valves s_2 and s_3 are closed and s_1 and s_4 are open. Furthermore, the fluid velocity entering the membrane unit, v_{fr}, is 10 $\frac{m}{s}$. When flow reversal is initiated, solenoid valves s_2 and s_3 are opened and assuming (for the purpose of this illustration) similar flow resistance in each path, the flow splits approximately evenly through the three possible

paths. The flow velocities through s_2, s_3, and the membrane unit are approximately 3.3 $\frac{m}{s}$. However, the fluid velocity through s_1 and s_4 is approximately 6.7 $\frac{m}{s}$. With a maximum velocity threshold of 1.5 $\frac{m}{s}$, the fluid velocities in these resulting streams are high enough to cause component damage through water hammer. From this example, it is clear that with the configuration seen in Fig. A.3, water hammer would be of concern when the fluid velocity into the membrane units is approximately 1.5 times the water hammer threshold.

Motivated by these considerations, the goal of this part of the thesis is to use model-predictive control (MPC) to determine the optimal switching path from normal operating conditions to a condition where the stream velocity entering the membranes is much lower; preventing water hammer during solenoid valve closure while avoiding pressure fluctuations and decreased process performance during the transition. Alleviating these phenomena will prolong equipment life-span and help to maximize the productivity of the RO system. The formulation presented in this work is specific to the system presented in Fig. A.3, but it is important to note that these MPC algorithms can be adapted to any flow-reversal equipped reverse osmosis system where the operator is able to control the stream velocity entering the membrane units (these algorithms could be utilized as a back-up configuration in the event of VFD failure or inoperability). Model-predictive control has not been employed for use with the feed flow reversal technique, but has been evaluated for the overall control of RO desalination processes [3]. It is noted that in a recent work [24], Lyapunov-based nonlinear

control systems and model-based monitoring schemes were designed for fault-tolerant control of RO processes in the presence of actuator faults without dealing with the issue of optimization of the feed flow reversal process (using a similar model to the system model presented in this work). Optimization with MPC requires the use of a RO desalination system model, which has been derived based on mass and energy balances (Chapter 3). A cost function that takes into account control action, stream velocities, and system pressure is proposed, along with several "hard" process constraints which represent physical limitations of the system. The model, cost function, and constraints are arranged into a non-linear optimization problem which is solved through the use of a numerical optimization algorithm. Closed-loop simulations with MPC are performed in this work to demonstrate the mode switching dynamics. The MPC algorithm is also applied to the system when a plant-model mismatch on feed concentration is imposed. This plant-model mismatch will allow for evaluation of the disturbance rejection capabilities of the controller when using feedback from the plant model.

RO System Model

Feed water enters the system and is pressurized by the high-pressure pump (Fig. A.3). The pressurized stream is split into a bypass stream (with velocity v_b) and the stream which enters the spiral-wound membrane unit(s) (v_{fr}). Two streams also exit the membrane module, the retentate (or brine) stream, with velocity v_r, and the

permeate stream. The downstream pressure of all of the exit streams is assumed to be atmospheric pressure.

This system set-up is a special case of the model derivation in Chapter 3 (in the case of the M3 system where $A_p^f = A_p^r = A_p^p = A_p$), occurring when the variable frequency drive on the feed pump is inoperable, and the feed flow rate is fixed (Fig. A.2). In this case, a bypass valve can be added to control the feed flow rate into the RO membrane units. The pressurized feed stream is split into a bypass stream (with velocity v_b) and the stream which enters the spiral-wound membrane unit(s) to adjust the velocity of feed water to the RO membrane units. An identical derivation (but with the addition of the bypass stream) to the one presented in Chapter 3 is conducted, and results in one additional differential equation for bypass stream velocity. It is noted that this is a specialized case (using a bypass valve instead of VFD on the feed pump) which may be practical for some small systems or systems where a VFD is either inoperable or can not be installed. The mass balance around the entire system, and the energy balances around each individual valve yield the following equations:

$$\frac{dv_b}{dt} = \frac{A_p^2}{A_m K_m V}(v_f - v_b - v_r) + \frac{A_p}{\rho V}\Delta\pi - \frac{1}{2}\frac{A_p e_{vb} v_b^2}{V} \tag{A.1}$$

$$\frac{dv_r}{dt} = \frac{A_p^2}{A_m K_m V}(v_f - v_b - v_r) + \frac{A_p}{\rho V}\Delta\pi - \frac{1}{2}\frac{A_p e_{vr} v_r^2}{V} \tag{A.2}$$
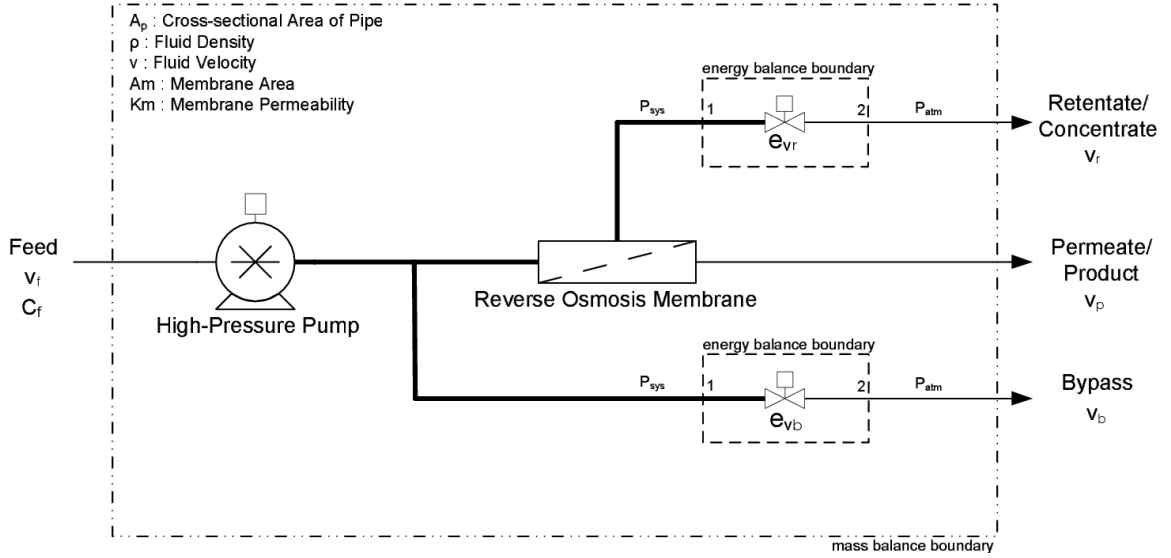
Figure A.2: RO desalination system equipped with bypass valve (instead of VFD) for RO module feed flow rate control.

$$P_{sys} = \frac{\rho A_p}{A_m K_m}(v_f - v_b - v_r) + \Delta\pi \qquad (A.3)$$

In order to accurately model the valve dynamics of the bypass and retentate valves, and also to obtain practical constraints, the concept of valve C_v is used. Depending on the type of valve and its flow characteristics, it is assumed that the C_v values (and in turn, the e_v values) can be related to the valve position (percentage open) through the empirical logarithmic relation presented in Chapter 3 (Eq. 3.21). For the model presented in this paper, the simplified curve relating valve position (O_p) to resistance value (e_v) is shown in Fig. A.4. The values of the constants μ and ϕ found in this empirical relation can be found in Table A.1.
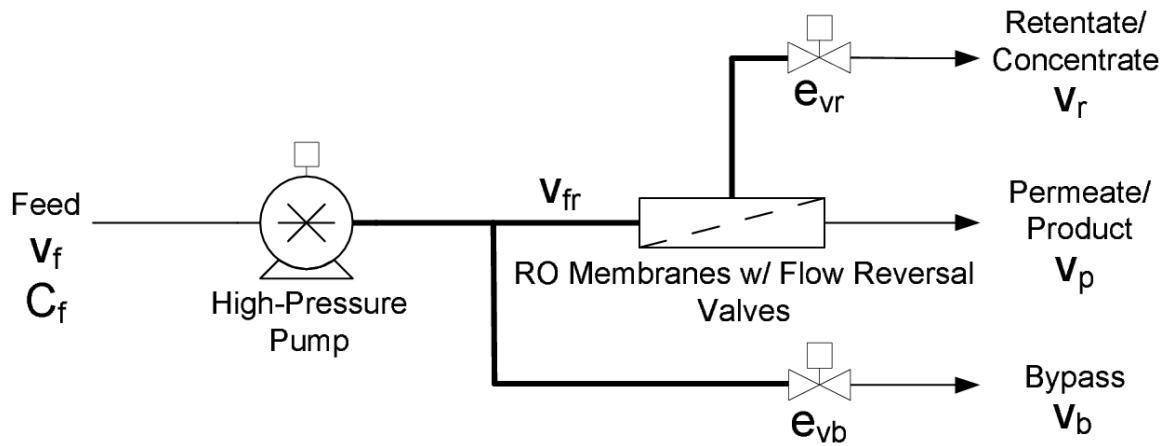
Figure A.3: Overall reverse osmosis system diagram.
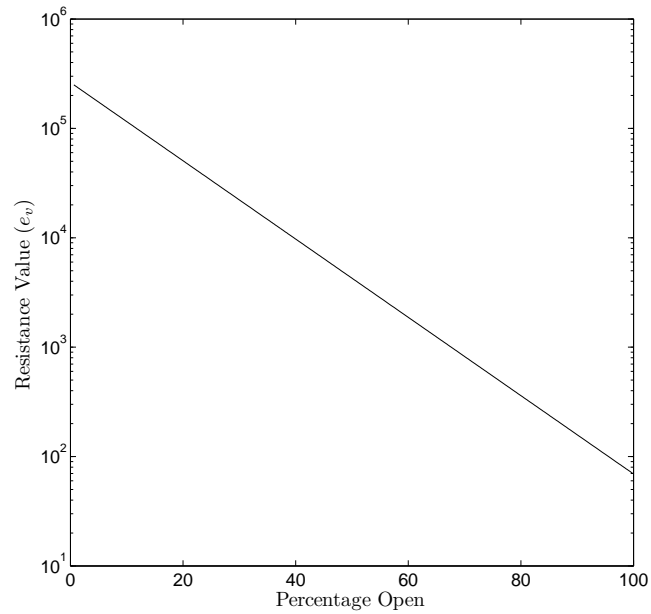
Figure A.4: Valve resistance values (e_v) vs. valve position.

Table A.1: Process parameters and normal mode steady–state values (nss).

ρ	=	1000	kg/m^3
V	=	0.04	m^3
v_f	=	10	m/s
A_p	=	1.27	cm^2
A_m	=	30	m^2
K_m	=	9.218×10^{-9}	s/m
C_f	=	10000	mg/L
a	=	0.5	
T	=	25	$^\circ C$
R	=	0.993	
μ	=	24.270	
ϕ	=	153.554	
δ	=	0.2641	$Pa/(ppm * K)$
v_b^{nss}	=	1.123	m/s
v_r^{nss}	=	4.511	m/s
P_{sys}^{sp}	=	457.51	psi
e_{vb}^{nss}	=	5000	
e_{vr}^{nss}	=	310	

Model Predictive Control Controller Formulation

When switching the system into flow-reversal mode, it is desired to bring the feed velocity into the membranes (v_{fr}) below the velocity threshold; where the flow will not cause significant water hammer when the solenoid valves are closed. In order to decrease the membrane feed velocity (v_{fr}), it will be necessary to open the bypass valve. It is desired to keep the system pressure constant while decreasing the velocity so that the membrane and system components will not be damaged. This can be done by closing the retentate valve while the bypass valve is being opened, in such a fashion that the system pressure fluctuates less than a pre-defined tolerance. MPC is used to complete this transition in an optimal way.

When reversing the flow, the solenoid valves (s_i), arranged as seen in Fig. A.1, are opened/closed in a specific sequence. First, valves s_2 and s_3 are opened, then valves s_1 and s_4 are closed. After these actions are completed, the retentate and bypass valves can be manipulated to return the process to the desired steady state. The normal steady state operating point is used as the initial condition for the mode switching. To determine the final state after mode switching, the procedure is as follows: Using the normal steady state operating point, the pressure set point is calculated using Eq. A.3. In this case, $P_{sys}^{sp} = 457.51$ psi. Second, setting the bypass velocity to $v_f - 1.5\frac{m}{s}$ (equivalent to setting $v_{fr} = 1.5\frac{m}{s}$) and using the desired pressure set point, the low-flow steady state operating point value for v_r^{lss} can be determined. With the

steady state values of v_r^{lss}, v_b^{lss}, and the pressure set point P_{sys}^{sp}, the model equations can be solved for the valve resistance values e_{vb}^{lss} and e_{vr}^{lss} corresponding to the low-flow operating point. Following this procedure, the low-flow steady state operating point is known, but the optimal path taken to get there from the initial operating steady state is not.

While it is desired to complete this flow direction switching with minimal impact on the system pressure, and in the shortest time possible, it is also necessary to factor in several system parameters such as pressure variation allowed, the bypass stream velocity as compared to the water hammer threshold, and the amount of control energy expended. To account for these issues, an optimization cost function is first proposed:

$$C(x, x_0, u) = \sum_{i=n_c}^{n_c+N} [\alpha(\frac{P_{sys}(i)}{P_{sys}^{sp}} - 1)^2 + \beta(\frac{v_{fr}(i)}{v_{wh}} - 1)^2 + \gamma((\frac{e_{vb}(i)}{e_{vb}^{lss}} - 1)^2 + (\frac{e_{vr}(i)}{e_{vr}^{lss}} - 1)^2)]$$

(A.4)

where n_c is the current time-step, $n_c + N$ is the current time-step plus the prediction horizon, α,β, and γ are dimensionless weighting coefficients, and v_{wh} is the water hammer threshold velocity. The prediction horizon, N, is defined such that the optimization is performed from the current time-step to N time-steps in the future (i.e., from $t = t_{current}$ to $t = t_{current} + N t_{step}$).

The values of the cost function of Eq. A.4 depend on the initial state of the

system (x_0) and the state of the system between $t(n_c)$ and $t(n_c + N)$ (the state, x, is comprised of v_b and v_r). The cost function also depends heavily on the control actions used (u), and weights given to the individual terms by the weighting coefficients α, β, and γ. As the optimization procedure is carried out, the optimization algorithm allows for a set of non-linear constraints to be employed. In this formulation, the following two hard actuator constraints are enforced:

$$O_{pi} \geq 0 \tag{A.5}$$

$$|\frac{dO_{pi}}{dt}| \leq R_{valve}^{max} \tag{A.6}$$

The first constraint forces the valve position values (O_{pi}) to be positive, since negative values of this variable would be physically meaningless. The second constraint sets a maximum rate of opening/closing for the valves, R_{valve}^{max}. Additional constraints can be added; constraints on maximum system pressure or other system variables may be desirable for certain types of RO operations.

In order to optimize the constrained transition from normal flow to low-flow and incorporate feedback into the calculation of the control action, a non-linear model-predictive control (MPC) formulation is implemented [27, 31, 65, 66]. In this method, a time frame for the transition is chosen, $t = 0$ to $t = t_f$, along with an optimization

time-step t_{step} and a prediction horizon N. Using these optimization parameters and the constraints along with the RO system model and the cost function weighting parameters α, β, and γ, the MPC control scheme can determine an optimal pair of control inputs, e_{vb} and e_{vr}, for each time-step.

The MPC optimization involves the following procedure:

1. The initial state vector and initial control value guesses are passed to a non-linear optimization algorithm based on sequential quadratic programming;

2. The optimization algorithm numerically integrates the model equations from $t = t_{current}$ to $t = t_{current} + N t_{step}$ using the initial state vector and control value guesses;

3. The resulting state vector is used to calculate the value of the cost function;

4. A new set of control inputs are determined, and steps 2-4 are repeated until a minimum cost value is found (i.e., $\min_{e_{vb},e_{vr}} C(x, x_0, u)$) subject to the constraints of Eqs. A.5 and A.6;

5. Optimal control inputs for each t_{step} are made available to the controller and actuators;

6. Only the first of the optimal control inputs, $u(t_{current})$, is applied; the system of Eqs. A.1 - A.3 is numerically integrated for one time-step (from $t = t_{current}$ to $t = t_{current} + t_{step}$) using the first optimal control value to yield a new initial state for the next optimization;

221

7. The remaining optimal control values for the prediction horizon are used as an initial guess for the computation of the control values in the next step;

8. All steps are repeated for each optimization time-step from $t = 0$ to $t = t_f - t_{step}$.

As implicitly stated above, all control values are applied in a sample-and-hold fashion; that is, a control value used in the integration on the interval $[t_{step_{n-1}} \ t_{step_n})$ is held constant over the entire interval, and then a new control value is determined by the optimization for the interval $[t_{step_n} \ t_{step_{n+1}})$.

Simulation Results

Overview

In order to test the feasibility of MPC for feed flow reversal in a reverse osmosis desalination system, several simulation studies were carried out. Initially, it was desired to examine the effect of using the model-predictive controller to switch between steady states when the process conditions are identical to the nominal plant model. Using a sampling time approximately one tenth of the system step response time, the model-predictive control formulation is applied to the system with various prediction horizons. These simulations are subsequently compared to an "open-loop manually controlled" transition where the control inputs are manipulated to their final values at the maximum rate allowed by the constraints, as well as the case where the transition is controlled using proportional-integral (PI) control.

Next, it was desired to simulate the switching between steady states in the presence of a plant-model mismatch on the feed TDS value. The controller receives state feedback from the plant model at the end of each time-step (i.e., measurements of v_b and v_r), but an offset in system pressure and stream velocity is observed due to the mismatched MPC controller. Simulations are conducted at several prediction horizons, and an integral control input is applied after the MPC reaches steady state in order to bring the system pressure back to the nominal pressure set point.

Finally, due to the relatively fast time scale of the system dynamics, the use of a steady-state approximation of the dynamic model equations (Eqs. A.1 - A.3) was also investigated in the non-linear MPC formulation. Simulation results evaluating the effectiveness of this approach are presented.

Optimal Mode Transition Without Plant-Model Mismatch

As described in the "Overview" section, the first simulations demonstrate the switching to low-flow mode using model-predictive control (MPC) in the case where the controller model and the plant model are identical. The controller uses measurements of the retentate and bypass stream velocities (system states) and manipulates the valve resistance values. In these simulations, the optimization parameters were set as shown in Table A.2. Additionally, the system was simulated for 10 seconds (t_f), and the prediction horizon, N, was varied in each simulation. The results are presented in Figs. A.5 - A.7.

Table A.2: Optimization parameters and low-flow mode steady–state values (lss).

t_0	=	0	s
t_{step}	=	0.1	s
α	=	10000	
β	=	100	
γ	=	200	
v_{wh}	=	1.5	m/s
v_b^i	=	1.123	m/s
v_r^i	=	4.511	m/s
C_{feed}^c	=	10000	ppm
R_{max}	=	10	$\%/s$
v_b^{lss}	=	8.5	m/s
v_r^{lss}	=	0.267	m/s
P_{sys}^{sp}	=	457.51	psi
e_{vb}^{lss}	=	87.322	
e_{vr}^{lss}	=	88592	
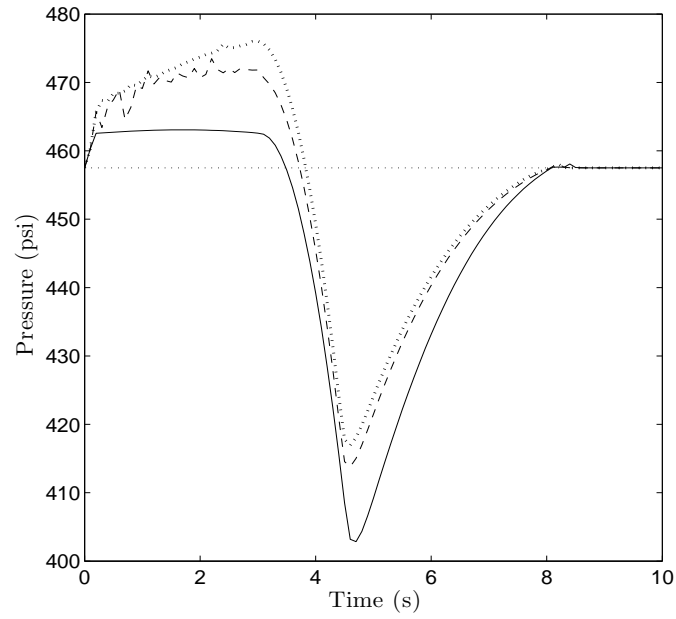
Figure A.5: Steady-state switching using MPC in the absence of plant-model mis-match: system pressure vs. time for N=1 (solid line), N=3 (dashed line), and N=5 (dotted line), including pressure set-point (horizontal line). Operating conditions for both steady states (beginning and end of transition) are given in Tables A.1 and A.2.
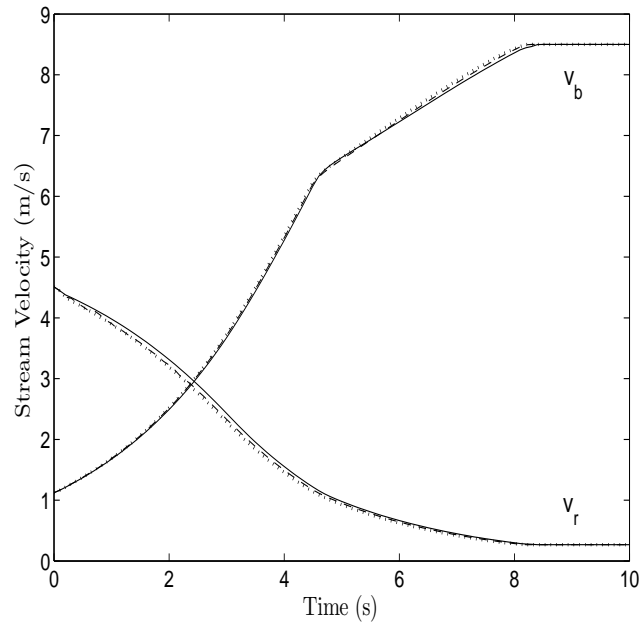
Figure A.6: Steady-state switching using MPC in the absence of plant-model mis-match: retentate and bypass stream velocities vs. time for N=1 (solid line), N=3 (dashed line), and N=5 (dotted line). Operating conditions for both steady states (beginning and end of transition) are given in Tables A.1 and A.2.
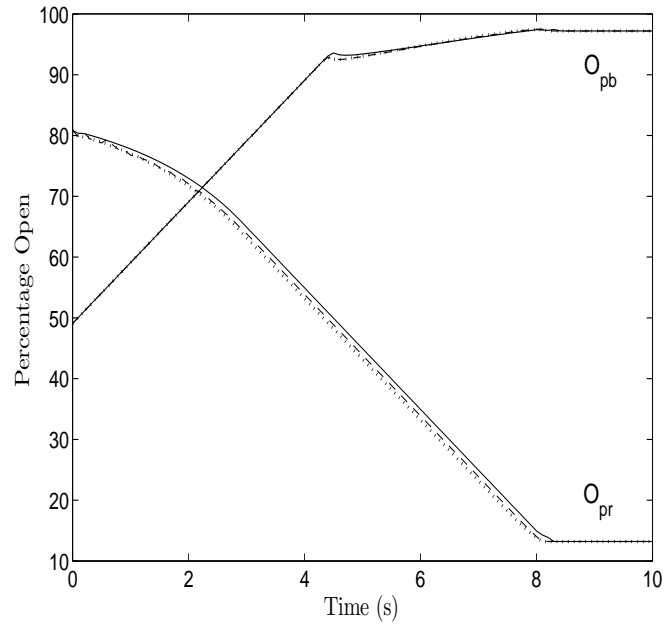
Figure A.7: Steady-state switching using MPC in the absence of plant-model mis-match: valve positions vs. time for N=1 (solid line), N=3 (dashed line), and N=5 (dotted line). Operating conditions for both steady states (beginning and end of transition) are given in Tables A.1 and A.2.

It can be seen that in the valve position and stream velocity plots (Figs. A.6 - A.7), only a small difference is observed between simulations with various prediction horizons. Even though the difference in control action is small, a large effect is seen on the system pressure, seen in Fig. A.5. In the case of the smallest prediction horizon ($N = 1$), the system pressure drops by approximately 55 psi before returning to the set point. It is seen that as the prediction horizon increases, the maximum deviation from the system pressure set point decreases, showing that the model-predictive control horizon is instrumental in minimizing pressure fluctuations.

The benefits of implementing MPC on the system pressure can be seen even more clearly when the optimized cases are compared to the "open-loop manually controlled" pressure in Fig. A.8, where the valves are adjusted to their final steady state at the maximum rate allowable by the constraints. In this case, a 100+ psi pressure variation caused by the "open-loop manually controlled" operation is observed; about two times larger than the one under MPC. Of course, the acceptable pressure deviation during mode transition depends on the specific RO process under consideration. However, the proposed MPC approach to addressing this control problem is flexible enough to allow for variation in the acceptable pressure level. Furthermore, it is important to point out that one can formulate "hard" constraints on the pressure in the optimization formulation of the MPC instead of penalizing the pressure deviation in the cost function (which is a "soft" constraint formulation) at the expense of restricting the feasibility region of the optimization problem.
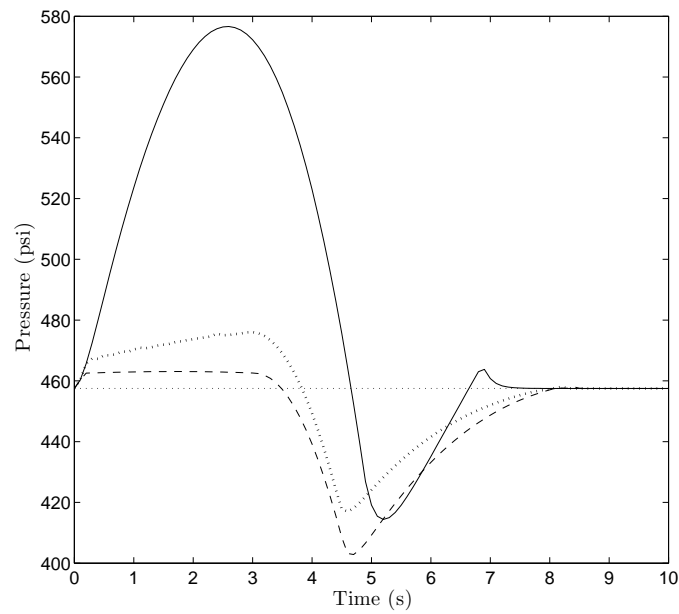
Figure A.8: Steady-state switching using MPC in the absence of plant-model mismatch: system pressure vs. time for "open-loop manually controlled" case (solid line), N=1 (dashed line), and N=5 (dotted line), including pressure set-point (horizontal line). Operating conditions for both steady states (beginning and end of transition) are given in Tables A.1 and A.2.

It was also desired to compare the performance of the MPC to proportional-integral (PI) control. Two PI loops were implemented, one loop measuring the bypass stream velocity and using the bypass valve resistance to bring the bypass stream velocity to the water hammer threshold, and another loop measuring the system pressure while adjusting the retentate valve to maintain the system pressure at the set point. These two loops can be represented as follows:

$$u^r_{PI} = K_r(P_{sys} - P^{sp}_{sys}) + \frac{1}{\tau_r} \int_0^{t_c} (P_{sys} - P^{sp}_{sys})dt \tag{A.7}$$

$$u^b_{PI} = K_b(v_b - v^{lss}_b) + \frac{1}{\tau_b} \int_0^{t_c} (v_b - v^{lss}_b)dt \tag{A.8}$$

Closed-loop simulations were carried out under various PI controller tunings in order to determine the best achievable closed-loop responses. The best achievable closed-loop responses under two different approaches are presented: in the first approach, the PI parameters ($K_r = -30$, $K_b = 1000$, $\tau_r = -30$, $\tau_b = 1000$) were chosen so that the transition is accomplished in a comparable amount of time to the MPC controlled case. It is observed that this case has a poor transient closed-loop performance, due to the presence of large oscillations. It is also noted that the integral term of the PI controller is switched off when the control action is saturated (reaches maximum rate constraint or valve position reaches 100%) to mitigate the effect of

integrator wind-up in the closed loop system. In the second tuning approach, the PI parameters ($K_r = -5$, $K_b = 800$, $\tau_r = -20$, $\tau_b = 500$) were chosen in order to conduct the fastest response that does not exhibit any oscillations during the transition between the original and final steady states. In this case, the pressure drops significantly more than any of the MPC cases (to 20 psi less than the first PI approach), and takes a much longer time to converge back to the steady state. The results can be seen in Figs. A.9 - A.10. The comparisons of MPC with PI demonstrate that under the MPC formulation, the pressure will deviate from the set point less than the PI controlled case regardless of the PI tuning parameters. The MPC also provides a smoother transition which is accomplished in less time.

In Fig. A.11, it can be observed that the values of the cost function decrease with increasing prediction horizon. The cost of these MPC controlled transitions fall between a lower and an upper bound; if the pressure weighting in the cost function is set to zero (that is, the pressure is allowed to deviate with no penalty) and transition speed becomes the only factor in switching steady states, then the valves will be opened and closed as fast as possible (equivalent to the "open-loop manually controlled" case). This situation leads to a lower bound on the achievable cost since all of the MPC controlled cases are penalized by pressure fluctuations. In the opposite situation, the MPC controlled cases should perform better than the maximum speed transition (again, the "open-loop manually controlled" case) where the pressure is weighted equivalently to the MPC controlled cases. It can be seen in Fig. A.11 that
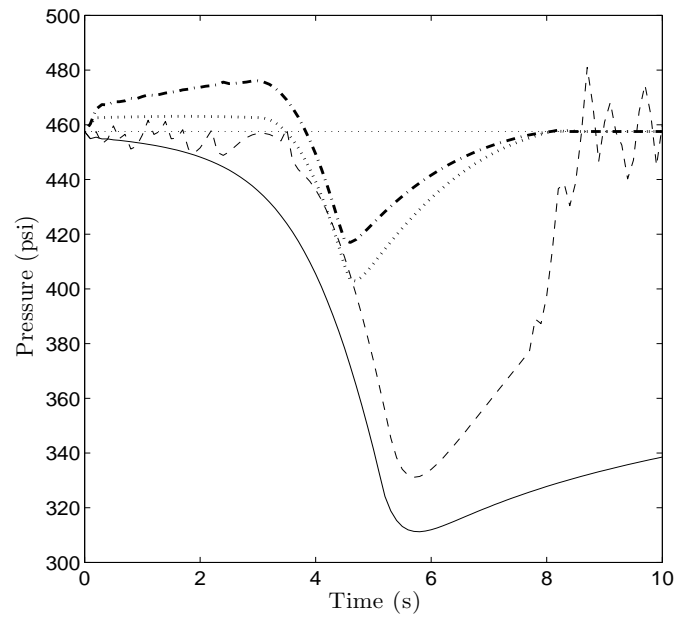
Figure A.9: Steady-state switching using MPC and PI in the absence of plant-model mismatch: pressure vs. time for first PI approach (dashed line), second PI approach (solid line), N=1 (dotted line), and N=5 (dash-dotted line), including pressure set-point (horizontal line). Operating conditions for both steady states (beginning and end of transition) are given in Tables A.1 and A.2.
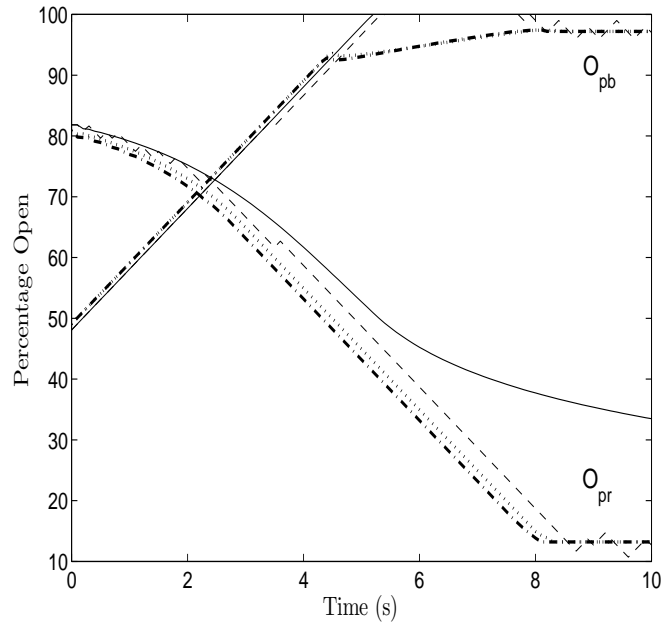
Figure A.10: Steady-state switching using MPC and PI in the absence of plant-model mismatch: valve positions vs. time for first PI approach (dashed line), second PI approach (solid line), N=1 (dotted line), and N=5 (dash-dotted line). Operating conditions for both steady states (beginning and end of transition) are given in Tables A.1 and A.2.
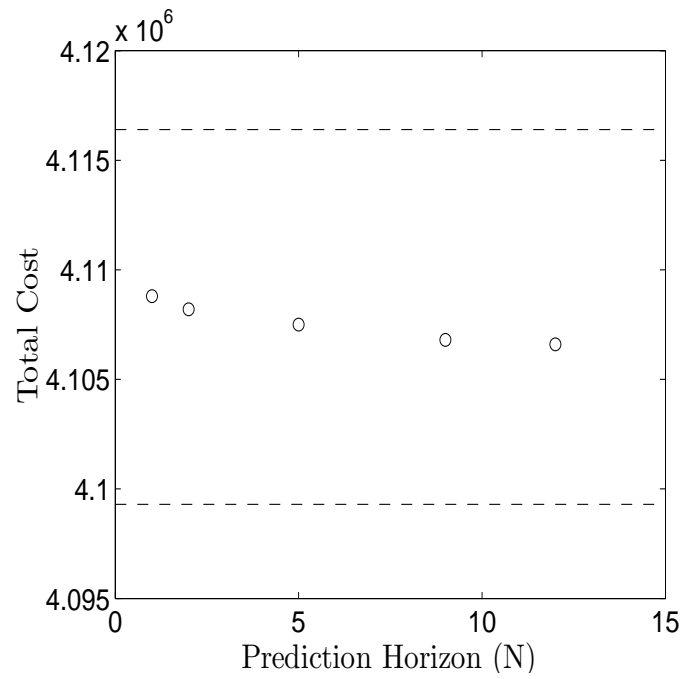
Figure A.11: Total optimization cost (Eq. A.4) vs. prediction horizon with upper and lower bounds based on maximum transition speed.

all of the MPC controlled cases at various prediction horizons fall between these two bounds. It is also noted that the magnitudes of the cost function values depend on the individual weighting on each term, but the trend will be independent of term weighting.

Optimal Mode Transition With Plant-Model Mismatch On Feed Quality

In order to evaluate the effectiveness of the MPC in the presence of a disturbance, a plant-model mismatch was imposed on the feed salinity (in terms of TDS value). In these simulations, the MPC algorithm continues to use the nominal feed TDS value of 10,000 ppm, while the plant simulation is conducted using the true feed value of 9,000 ppm.

It can be seen in Fig. A.12 that this plant-model mismatch results in a sizeable pressure offset from the original set-point. This offset is due to the fact that the controller is using the nominal feed TDS value in its calculations instead of the actual plant value. An offset is also observed in the velocity and control action plots (Figs. A.13 - A.14), but to a lesser degree. The magnitude of the offset also changes when the prediction horizon is varied, due to the increased performance of the optimization algorithm when the prediction horizon is increased. If it is necessary to bring the system pressure back to the nominal set point, integral control can be applied to the system after the MPC has completed its task. This integral term can be implemented in the following two ways: First,
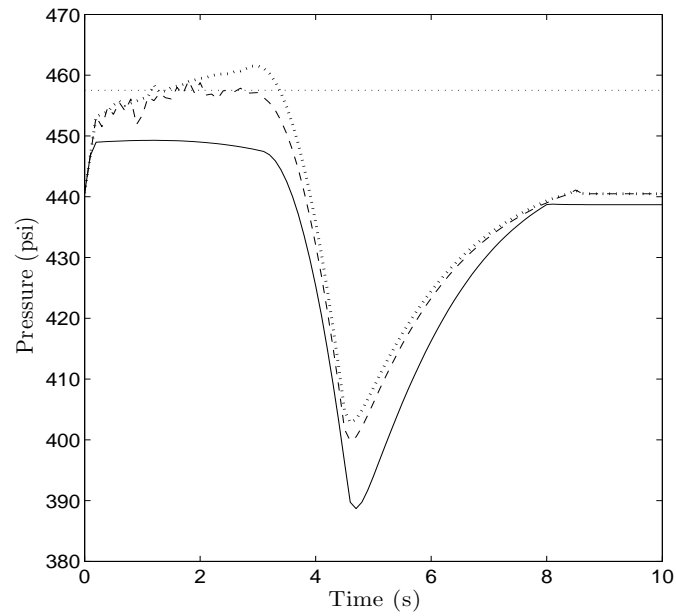
Figure A.12: Steady-state switching using MPC in the presence of plant-model mismatch on feed TDS: system pressure vs. time for N=1 (solid line), N=3 (dashed line), and N=5 (dotted line), including pressure set-point (horizontal line). Operating conditions for both steady states (beginning and end of transition) are given in Tables A.1 and A.2.
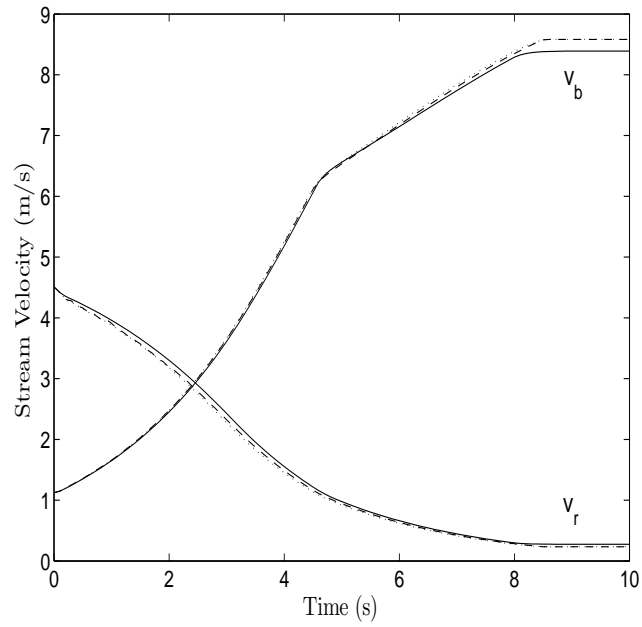
Figure A.13: Steady-state switching using MPC in the presence of plant-model mismatch on feed TDS: stream velocities vs. time for N=1 (solid line), N=3 (dashed line), and N=5 (dotted line). Operating conditions for both steady states (beginning and end of transition) are given in Tables A.1 and A.2.
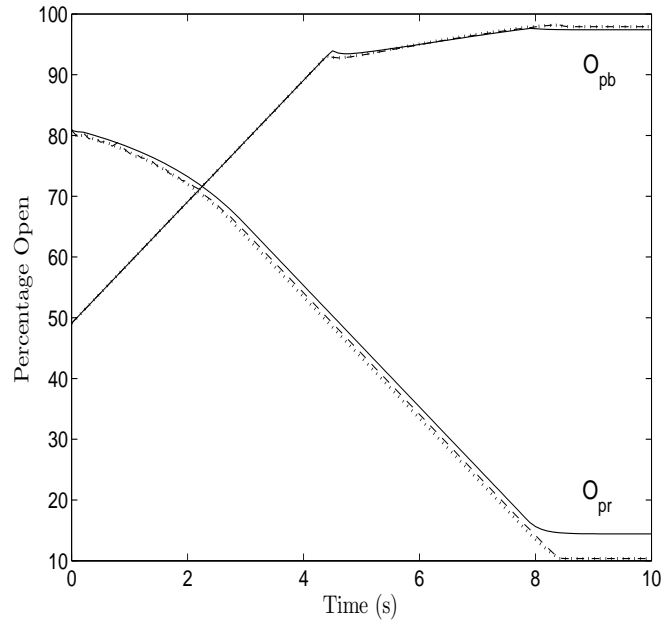
Figure A.14: Steady-state switching using MPC in the presence of plant-model mismatch on feed TDS: valve positions vs. time for N=1 (solid line), N=3 (dashed line), and N=5 (dotted line). Operating conditions for both steady states (beginning and end of transition) are given in Tables A.1 and A.2.

$$u_{total}^r = u_{MPC}^{rf} + \frac{1}{\tau_i^r} \int_{10}^{t_c} (P_{sys} - P_{sys}^{sp}) dt \qquad \text{(A.9)}$$

where u_{MPC}^{rf} represents the final value for the MPC retentate valve control action (in terms of valve % open) after reaching the steady state determined by the optimization, t_c is the current time (in seconds), and $(P_{sys} - P_{sys}^{sp})$ (in psi) is the error between the actual system pressure and the nominal set point pressure, and τ_i^r is the integral time constant ($\tau_i^r = 10\ s$). Second,

$$u_{total}^b = u_{MPC}^{bf} + \frac{1}{\tau_i^b} \int_{10}^{t_c} (v_b - v_b^{lss}) dt \qquad \text{(A.10)}$$

where u_{MPC}^{bf} (in terms of valve % open) represents the final value for the MPC bypass valve control action after reaching the steady state determined by the optimization, $(v_b - v_b^{lss})$ is the error between the actual bypass velocity and the low-flow steady state bypass velocity (in m/s), and τ_i^b is the integral time constant ($\tau_i^b = \frac{1}{30}\ s$).

In the results presented in Figs. A.15 - A.17, it is seen that the MPC optimization reaches a steady state around $t = 8$ seconds; after this steady state is reached, the MPC is deactivated and the integral control is initiated at $t = 10$ seconds. In both cases, the offsets are eliminated; the offset on pressure is eliminated in the case of the integral term using pressure measurements, and the offset on bypass velocity is eliminated in the second case. It can also be observed that the system pressure

239
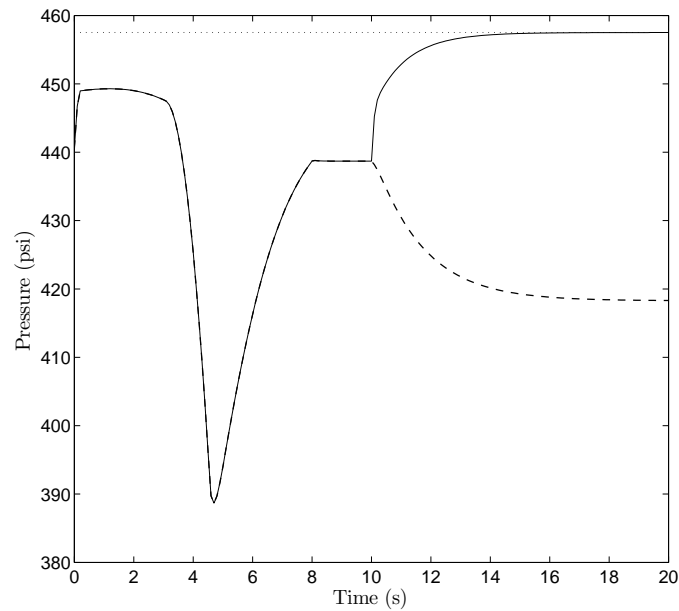
Figure A.15: Mode transition using MPC and integral control in the presence of plant-model mismatch on feed TDS: system pressure vs. time for integral term based on system pressure (solid line) and integral term based on bypass velocity (dashed line) with pressure set-point (dotted line) for N=1. Operating conditions for both steady states (beginning and end of transition) are given in Tables A.1 and A.2.

deviates even more than the original offset in the case where the integral term is based on the bypass velocity. Other methods may be used to correct for MPC offset due to plant-model mismatch; the approach followed here is only one example. For more detailed information on PI controller tuning, see [64].

240
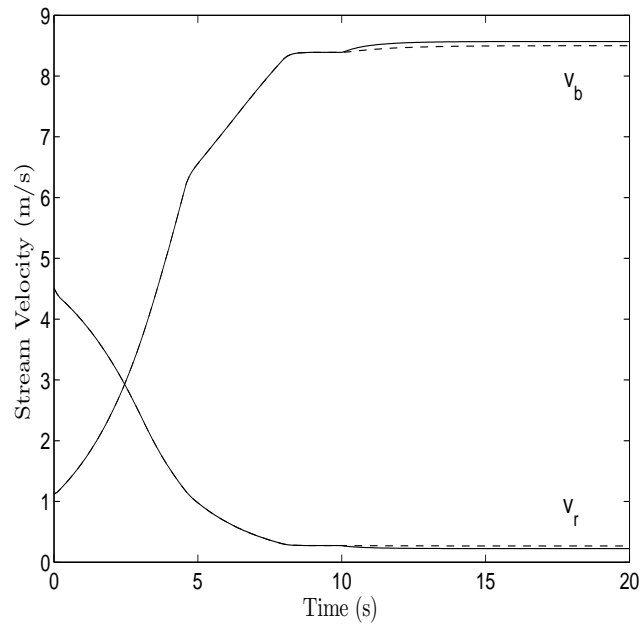
Figure A.16: Mode transition using MPC and integral control in the presence of plant-model mismatch on feed TDS: bypass and retentate stream velocities vs. time for integral term based on system pressure (solid lines) and integral term based on bypass velocity (dashed lines) for N=1. Operating conditions for both steady states (beginning and end of transition) are given in Tables A.1 and A.2.
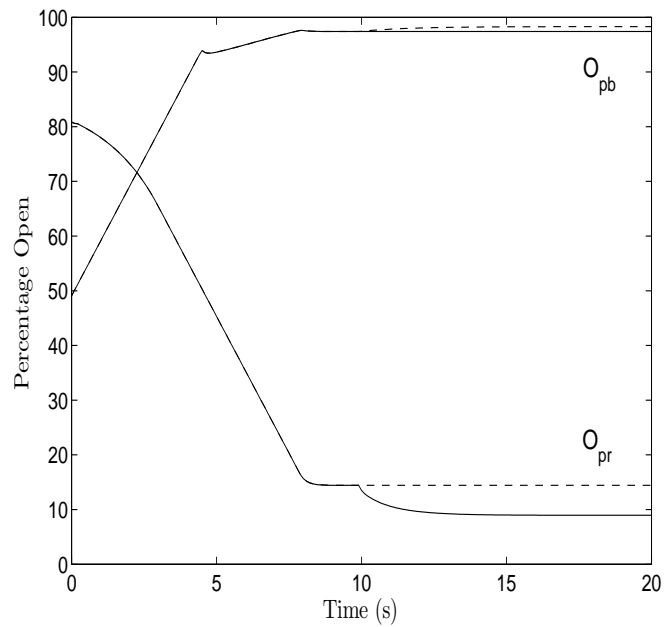
Figure A.17: Mode transition using MPC and integral control in the presence of plant-model mismatch on feed TDS: valve positions vs. time for integral term based on system pressure (solid lines) and integral term based on bypass velocity (dashed lines) for N=1. Operating conditions for both steady states (beginning and end of transition) are given in Tables A.1 and A.2.

Use of Steady-State Process Model in MPC

Specifically, motivated by the fast time-scale of the RO system dynamics, an algebraic steady-state approximation of the model equations (Eqs. A.1-A.3) is used in place of the dynamic model in the MPC formulation. In this case, the MPC algorithm with a prediction horizon N=1 takes the form:

$$
\min_{e_{vb},\, e_{vr}} \sum_{i=n_c}^{n_c+1} [\alpha(\frac{P_{sys}(i)}{P_{sys}^{sp}} - 1)^2 + \beta(\frac{v_{fr}(i)}{v_{wh}} - 1)^2 + \gamma((\frac{e_{vb}(i)}{e_{vb}^{lss}} - 1)^2 + (\frac{e_{vr}(i)}{e_{vr}^{lss}} - 1)^2)]
$$

$$
0 = \frac{A_p^2}{A_m K_m V}(v_f - v_b - v_r) + \frac{A_p}{\rho V}\Delta\pi - \frac{1}{2}\frac{A_p e_{vb} v_b^2}{V}
$$

$$
0 = \frac{A_p^2}{A_m K_m V}(v_f - v_b - v_r) + \frac{A_p}{\rho V}\Delta\pi - \frac{1}{2}\frac{A_p e_{vr} v_r^2}{V}
$$

$$
O_{pi} \geq 0, \; |\frac{dO_{pi}}{dt}| \leq R_{valve}^{max}
$$

$$
\Delta\pi = \delta C_{eff}(T + 273)
$$

$$
C_{effective} = a C_{feed} + (1 - a) C_{retentate}
$$

$$
P_{sys} = \frac{\rho A_p}{A_m K_m}(v_f - v_b - v_r) + \Delta\pi
$$

$$\tag{A.11}$$

In Figs. A.18 - A.20, the closed-loop system results (under the MPC) using the steady-state algebraic equations with a sampling time of 1 s are presented and compared to the "open-loop manually controlled" case (where the valves are opened to their final steady-state at the maximum rate allowed).

In the stream velocity and control action plots (Figs. A.19 - A.20), it is seen that the algebraic steady-state MPC formulation is very similar to the "open-loop

manually controlled" case, but the small differences in control action have a large effect on the system pressure. A large fluctuation in system pressure (approximately 125 psi) can be observed in Fig. A.18, showing that even with a larger sampling time, a prediction horizon of one, and only using steady-state algebraic equations (Eqs. A.11) in the optimization algorithm, the MPC formulation still performs much better than the "open-loop manually controlled" case in terms of pressure fluctuations (the pressure deviation in the MPC case is only about 25 psi, as compared to 125 psi in the "open-loop manually controlled" case and PI approach). As expected, the steady-state MPC formulation does not perform as well as the closed-loop system using the MPC with dynamic model equations. In terms of computation time, it was found that the steady-state algebraic MPC formulation and the MPC formulation using the dynamic model computed the optimal control actions in approximately the same time (for any given prediction horizon). The steady-state algebraic MPC formulation of Eq. A.11 is slightly faster in terms of computation time relative to the MPC with the dynamic process model, but the benefits of increased performance in the dynamic formulation far outweigh this discrepancy in computation time. This result demonstrates that the majority of computation time is used to perform the optimization, and not to perform the integration of the dynamic system model in the controller. Overall, it is more beneficial to use the dynamic model in the closed-loop MPC algorithm with a dedicated processor to carry out the calculations.
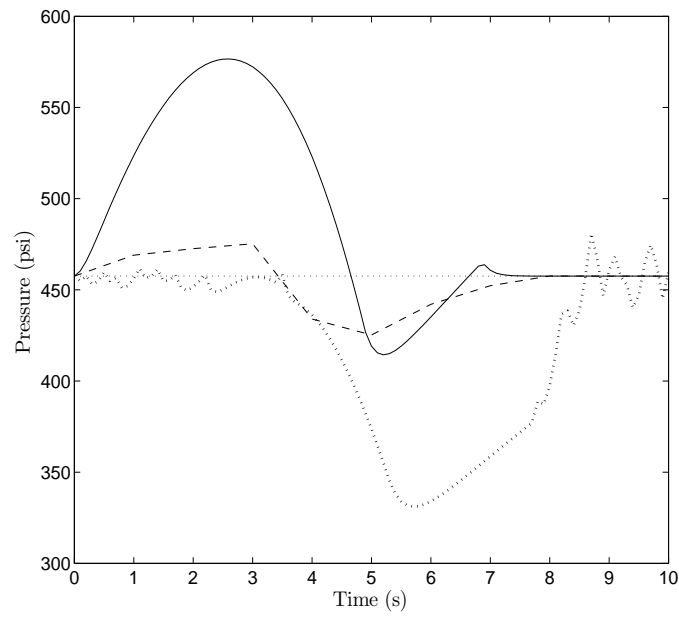
Figure A.18: Mode transition using MPC in the absence of plant-model mismatch: system pressure vs. time for algebraic steady-state MPC formulation (dashed line), "open-loop manually controlled" case (solid line), and first PI approach (dotted line). Operating conditions for both steady states (beginning and end of transition) are given in Tables A.1 and A.2.
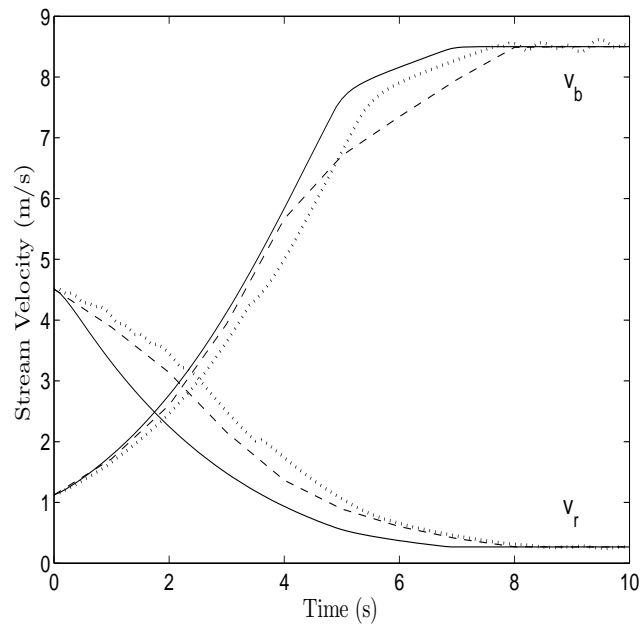
Figure A.19: Mode transition using MPC in the absence of plant-model mismatch: stream velocities vs. time for algebraic steady-state MPC formulation (dashed line), "open-loop manually controlled" case (solid line), and first PI approach (dotted line). Operating conditions for both steady states (beginning and end of transition) are given in Tables A.1 and A.2.
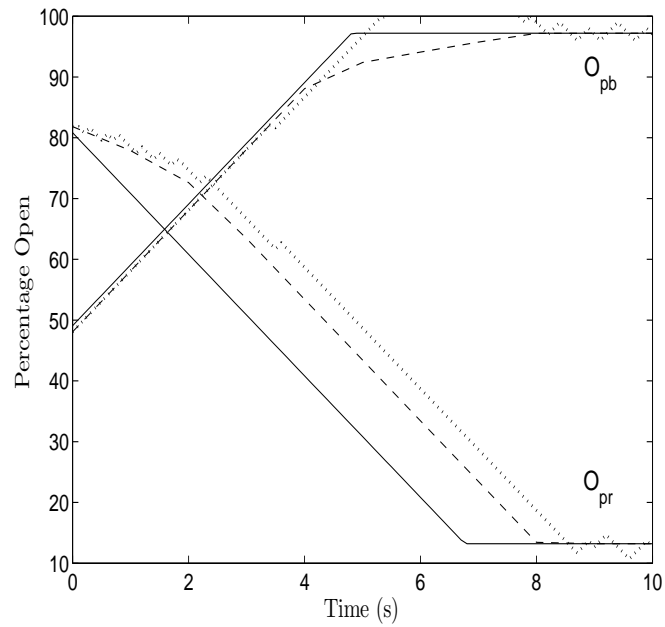
Figure A.20: Mode transition using MPC in the absence of plant-model mismatch: valve positions vs. time for algebraic steady-state MPC formulation (dashed line), "open-loop manually controlled" case (solid line), and first PI approach (dotted line). Operating conditions for both steady states (beginning and end of transition) are given in Tables A.1 and A.2.

Conclusions

A model-predictive control strategy was developed for switching between the normal flow operating steady state and the feed flow reversal steady state (such that water hammer is prevented when solenoid valves are closed). First, a dynamic model of the process was developed as a function of the process parameters, feed concentration, and the bypass/retentate valve resistance values. Using these valve resistance values as control inputs, a non-linear optimization problem was formulated. Solving this optimization through a model-predictive control framework, it was shown that a feedback-based controller allowed the system to make the transition between steady states with a much smaller variation in system pressure (approximately 25 psi for the MPC case as compared to 150 psi for the open-loop case). The MPC framework was also shown to have smaller pressure fluctuations and shorter transition time than several well-tuned PI controllers. Non-linear MPC was also shown to be beneficial in the presence of plant-model mismatch. The feedback-based MPC algorithm also improved the speed at which the stream velocities reached the feed flow reversal steady state, decreased the offset between the actual final steady-state and the desired final steady-state, and damped oscillations in the control action. It was also demonstrated that the benefits of using the dynamic MPC formulation to provide increased system performance outweighed the slightly decreased computation time of using the MPC with a steady-state process model.

B   M3/CoM2RO System Sensors, Actuators, and Control System Hardware

Table B.1: M3 system sensors and actuators.

Component	Manufacturer	Model	Quantity
Flow sensor	GF Signet	2100	2
Flow sensor	GF Signet	2537	3
Flow transmitter	GF Signet	8550	2
Conductivity sensor	GF Signet	2850	4
Conductivity probe	GF Signet	2842	2
Conductivity probe	GF Signet	2840	2
pH sensor/electrode	GF Signet	8750	2
Pressure sensor	Omega	PX409-1.0KG5V	2
Pressure sensor	Omega	PX409-100G5V	2
Temperature sensor	Wika	05-97	1
Voltage sensor	Magnelab	SRT-0375-150 H476	5
Current sensor	CR Magnetics	CR8448-2000	20
Variable frequency drives	TECO	FM50	2
Actuated valve	ETI Systems	VA8V-10-0-10	1

Table B.2: M3 system control/DAQ hardware.

Component	Manufacturer	Model	Quantity
Stand-alone chassis/FPGA	National Instruments	cRIO-9104	1
Stand-alone RTC	National Instruments	cRIO-9014	1
-10-10V analog input module	National Instruments	9205	2
4-20mA analog output module	National Instruments	9265	1
-10-10V analog output module	National Instruments	9263	1
Relay module (electro-mech.)	National Instruments	9481	3

Table B.3: CoM2RO system sensors and actuators.

Component	Manufacturer	Model	Quantity
Flow sensor (magmeter)	GF Signet	3-2551-W0-12	7
Flow sensor (turbine)	Seametrics	SPX-075-13-07	2
Pressure sensor	AST	AST4000-A-0xxxx-P-4-E-1	11
Temperature sensor	GF Signet	2350-3	2
Turbidity sensor	GF Signet	4150	1
Conductivity sensor (reg.)	GF Signet	3-2850-52-41	3
Conductivity sensor (toroidal)	Sensorex	TCS3020-P1K + TCSMA	2
pH/ORP sensor	GF Signet	2750	3
Variable frequency drive (UF)	Danfoss	FC202P4K0T4E66H1	1
Variable frequency drive (RO)	Danfoss	FC202P22KT4E66H1	1
Actuated retentate valve	Jordan Valve	708LMO-075-S6	1
Actuated 2-way valve	GF Signet	199.107.207	8
Actuated 2-way valve	KZ Valve	F2H22-1ADY0-P01	6
Actuated 3-way valve	Plast-o-matic	TEBVA6-1-200EPS-PV-A	8
Actuated 3-way valve	KZ Valve	F4H22-1ADY0-P01	1
Actuated 2-way valve	KZ Valve	SRC Standard	1
Actuated solenoid valve	ASCO	8256A104E-24VDC	1
Actuated solenoid valve	Plast-o-matic	EAST4V8W11-PV	1
Self-cleaning filter	Amiad	TAF-500	1
Metering pump	Iwaki	EWB21Y1-VC	1

Table B.4: CoM2RO system control/DAQ hardware.

Component	Manufacturer	Model	Quantity
Integrated chassis/RTC (RO/UF)	National Instruments	cRIO-9074	2
Stand-alone chassis/FPGA (Sup.)	National Instruments	cRIO-9114	1
Stand-alone RTC (Sup.)	National Instruments	cRIO-9022	1
4-20mA analog input module	National Instruments	9408	2
4-20mA analog output module	National Instruments	9265	2
24VDC sourcing digital I/O module	National Instruments	9426	3
24VDC sinking digital I/O module	National Instruments	9421	1
Relay module (solid state)	National Instruments	9485	5
Relay module (electro-mech.)	National Instruments	9481	1

C   MATLAB Code for MPC of FFR, Energy Optimal Control, and Image Analysis

Main Code File for Model Predictive Control of Feed Flow Reversal ("FR_MPC.m")

```
%MPC FR Optimization
clear
options= optimset('LargeScale','off','MaxFunEvals',20000,'MaxIter',200);


global x0 t0 tstep N tf h Psp alpha beta gamma delta Rmax h
global Am Km rho vol Ap T B a R v1 Cfeed_actual Cfeed_controller ev1 ev2 ev1i ev2i ev1e ev2e
global xinitial xinitial_d tcurrent uinitial v_bypass v_ret xinit_OL lookup_table neg_count Rmax1
    Rmax2 totalcost


%----Optimization Parameters----
t0=0;
tf=10; %final time (0 is initial) (s)
tstep=1; %time steps for optimization
N=2; %prediction horizon (timesteps)
%h=0.05; %differential equation solver timestep
alpha=1e4; %weighting on system pressure
beta=100; %weighting on velocity
gamma=200; %weighting on control action
%-----------------


%--------System Parameters---------
Am=30; %m^2 membrane surface area
Km=9.218e-9; %s/m membrane permeability
```

```
rho=1000; %kg/m^3 water density

vol=0.04; %m^3 system volume 0.04

Ap=1.26e-4; %m^2 pipe cross-sectional area

T=25; %temperature (C)

R=0.993; %membrane fractional salt rejection

%--------------------------------


%--------Manipulated Constants---------

v1=10; %m/s feed flow velocity

Cfeed_actual=10000; %ppm initial feed concentration

Cfeed_controller=10000;

a=0.5; %weighting parameter for effective concentration (Ceff = a*Cf + (1-a)*Cb)

B=1; %weighting parameter for bulk vs. surface salt concentration in membrane

    %(dPi=0.2641*B*Ceff*(T+273))

ev1=5000; %main bypass valve constant

ev2=310; %retentate valve constant

ev1i=ev1; %main bypass valve constant

ev2i=ev2; %retentate valve constant

%------------------------------------


%--------Initial Steady State-------

xinitial=[1.1233;4.5107]; %from model equations

Ceff=Cfeed_controller*(a+(1-a)*((1-R)+(R*(v1-xinitial(1)))/(xinitial(2))));

dpi=0.2641*Ceff*(T+273);

Psp=((Ap*rho)/(Am*Km))*(v1-xinitial(1)-xinitial(2))+dpi;

Psp2=Psp/6895

%---------------------------------


%--------Determining Retentate Velocity---------

v_bypass=v1-1.5; %m/s

vr_guess=1;

v_ret=fsolve(@retentatevelocity,vr_guess) %determining vr for obtaining same Psys
```

```
%-----------------------------------------------


%--------Determining ev values----------
ev_guess=[310;5000];
ev_actual=fsolve(@evsteadystate,ev_guess);


ev1e=ev_actual(1) %ending steady state valve constant (to keep steady system pressure)
ev2e=ev_actual(2) %ending steady state valve constant
%------------------------------------


tspan_MPC=[0:tstep:tf]; %overall optimization length
tcurrent=0;
X_final=[];
X_final_d=[];
ui=[ev1i;ev2i];
U=ui;
control_action=[ui];
xinit_OL=xinitial;


%guessing control values for each timestep (linear)
for j=1:length(tspan_MPC)+N
    if j<=(length(tspan_MPC))
        uinitial_overall(1,j)=ev1e;%ev1i-((ev1-ev1e)/(tf-t0))*(tspan_MPC(j)-t0);
        uinitial_overall(2,j)=ev2e;%ev2i-((ev2-ev2e)/(tf-t0))*(tspan_MPC(j)-t0);
%        uinitial_overall(1,j)=ev1i-((ev1-ev1e)/(tf-t0))*(tspan_MPC(j)-t0);
%        uinitial_overall(2,j)=ev2i-((ev2-ev2e)/(tf-t0))*(tspan_MPC(j)-t0);
    end
    if j>(length(tspan_MPC))
%        uinitial_overall(1,j)=uinitial_overall(1,j-1);
%        uinitial_overall(2,j)=uinitial_overall(2,j-1);
        uinitial_overall(1,j)=ev1e;
        uinitial_overall(2,j)=ev2e;
```

```
        end

end


%Valve percentage-open lookup table


ev_list=[[10:1:100],[105:5:500],[510:10:2000],[2100:100:15000],[15500:500:50000],[51000:1000:250000]];

for table_index=1:length(ev_list)

    valve_percent_table(table_index)=-12.1349378*log(ev_list(table_index))+151.4420343;

end

lookup_table=[ev_list' valve_percent_table'];


%Valve perc - ev constants

a_ev=-12.1349378;

b_ev=151.4420343;


tic


for i=1:length(tspan_MPC)


    tic


    %recording actual and disturbed states

    X_final=[X_final,xinitial];


    tf=(N*tstep)+tcurrent;

    tspan=[tcurrent:tstep:tf];


    uinitial=uinitial_overall(:,[i:i+N]);

    uinitial(:,1)=ui;

    if i>1

        uinitial(:,[1:N])=ui2;

    end
```

```
%xinitial_d=xinit_OL;

xinitial_d=xinitial; %with correct feedback


perc_open=10;
%    perc_1_current=a_ev*log(U(1,1))+b_ev;

%    Rmax1=exp(((perc_1_current-(sign(ev1e-U(1,1))*perc_open*tstep))-b_ev)/a_ev);

%    Rmax1=abs(U(1,1)-Rmax1)

%

%    perc_2_current=a_ev*log(U(2,1))+b_ev;

%    Rmax2=exp(((perc_2_current-(sign(ev2e-U(2,1))*perc_open*tstep))-b_ev)/a_ev);

%    Rmax2=abs(U(2,1)-Rmax2)


control_action(1,i);

perc_1_current=a_ev*log(control_action(1,i))+b_ev;

Rmax1i=exp(((perc_1_current-(sign(ev1e-control_action(1,i))*perc_open*tstep))-b_ev)/a_ev);

Rmax1=abs(control_action(1,i)-Rmax1i);


control_action(2,i);

perc_2_current=a_ev*log(control_action(2,i))+b_ev;

Rmax2i=exp(((perc_2_current-(sign(ev2e-control_action(2,i))*perc_open*tstep))-b_ev)/a_ev);

Rmax2=abs(control_action(2,i)-Rmax2i);


uinitial

%Finding optimum control for all of prediction horizon

U = fmincon('FR_costfunction2',uinitial,[],[],[],[],50,100000,'FR_constraints2',options); %


%Taking first move from optimum values

ui=U(:,2);


%Recording valve % open

perc_1=a_ev*log(ui(1,1))+b_ev;
```

```matlab
        perc_2=a_ev*log(ui(2,1))+b_ev;

        valve_percentage(:,i)=[perc_1;perc_2];


        ui2=U(:,[2:end]);


        %Integrating ODEs ahead using control value
        %Cfeed=11000;
        [tsim,xsim] = ode23s(@ROsystemODEsOPT,[tcurrent (tcurrent+tstep)],xinitial,[],ui);
        %Cfeed=10000;


        %New initial state
        xsim=xsim';
        xinitial=xsim(:,end);


        X_final_d=[X_final_d,xinitial_d];


        if i<length(tspan_MPC)

            control_action=[control_action,ui];

            tcurrent=(tstep*i)

        end


        U
        opt_time=toc


end


totalcost=0;
for j=1:length(X_final)

    Ceff=Cfeed_actual*(a+(1-a)*((1-R)+(R*(v1-X_final(1,j)))/(X_final(2,j))));

    dpi=0.2641*Ceff*(T+273);

    P(j)=((Ap*rho)/(Am*Km))*(v1-X_final(1,j)-X_final(2,j))+dpi;
```

```
    %overall cost

    totalcost=totalcost+(alpha*((P(j)/Psp)-1)^2)+beta*((((v1-X_final(1,j))/1.5)-1)^2)+..

    (gamma*(((control_action(1,j)/ev1e)-1)^2)+

    (((control_action(2,j)/ev2e)-1)^2));


    P(j)=(P(j)/6895);
end


totalcost


for j=1:length(X_final_d)

    Ceff=Cfeed_actual*(a+(1-a)*((1-R)+(R*(v1-X_final_d(1,j)))/(X_final_d(2,j))));

    dpi=0.2641*Ceff*(T+273);

    P_d(j)=((Ap*rho)/(Am*Km))*(v1-X_final_d(1,j)-X_final_d(2,j))+dpi;

    P_d(j)=(P_d(j)/6895);
end


set(0,'DefaultTextInterpreter','latex')


%---System Pressure---
syspressurefig=figure;
set(syspressurefig,'name',['OPT: System Pressure (Open Loop) (alpha/beta = ',num2str(alpha/beta),

    ', N=',num2str(N),' )'],

'color', [1 1 1],'NumberTitle','off');
plot(tspan_MPC,P,'k-')
% hold on
% plot(tspan_MPC,P_d,'r-')
%syspressuretitle=title('System Pressure vs. Time');
%set(syspressuretitle,'FontSize',14);
xlabel('Time (s)');
ylabel('Pressure (psi)');
axis1=axis;
```

```
pspline=line([axis1(1),axis1(2)],[Psp2,Psp2]);

set(pspline,'linestyle',':','color','k');


%---Stream Velocity---

velocityfig=figure;

set(velocityfig,'name',['OPT: Stream Velocities (Open Loop) (alpha/beta = ',num2str(alpha/beta),

    ', N=',num2str(N),' )'],

'color', [1 1 1],'NumberTitle','off');

plot(tspan_MPC,X_final(1,:),'k-')

hold on

plot(tspan_MPC,X_final(2,:),'k--')

%plot(tspan_MPC,X_final_d(1,:),'r:')

%plot(tspan_MPC,X_final_d(2,:),'m:')

%vlegend=legend('Bypass ($v_b$)','Retentate ($v_r$)');%,'Noisy States - Bypass',

    'Noisy States - Retentate','Location',

%'NorthEastOutside');

%velocitytitle=title('Stream Velocities vs. Time');

%set(velocitytitle,'FontSize',14);

%set(vlegend, 'interpreter', 'latex');

xlabel('Time (s)');

ylabel('Stream Velocity (m/s)');


%---Control Action---

controlfig=figure;

set(controlfig,'name',['OPT: Control Action (Open Loop) (alpha/beta = ',num2str(alpha/beta),

    ', N=',num2str(N),',

 Rmax=',num2str(Rmax),' )'],'color', [1 1 1],'NumberTitle','off');

plot(tspan_MPC,control_action(1,:),'k')

hold on

plot(tspan_MPC,control_action(2,:),'k--')

%clegend=legend('Bypass Valve ($e_{vb}$)','Retentate Valve ($e_{vr}$)','Location','Best');

%syspressuretitle=title('Control Action vs. Time');
```

```
%set(syspressuretitle,'FontSize',14);

%set(clegend, 'interpreter', 'latex');

xlabel('Time (s)');

ylabel('Control Action');


%---Valve %---

percentfig=figure;

set(percentfig,'name',['OPT: Valve Open-Percentage (Open Loop) (alpha/beta = ',num2str(alpha/beta),

    ', N=',num2str(N),',Rmax=',num2str(Rmax),' )'],'color', [1 1 1],'NumberTitle','off');

plot(tspan_MPC,valve_percentage(1,:),'k')

hold on

plot(tspan_MPC,valve_percentage(2,:),'k--')

%clegend=legend('Bypass Valve ($e_{vb}$)','Retentate Valve ($e_{vr}$)','Location','Best');

%syspressuretitle=title('Control Action vs. Time');

%set(syspressuretitle,'FontSize',14);

%set(clegend, 'interpreter', 'latex');

xlabel('Time (s)');

ylabel('Percentage Open');
```

Cost Function File for Model Predictive Control of Feed Flow Reversal ("FR_costfunction2.m")

```
function C = FR_costfunction2(U)


global x0 t0 tstep N tf Psp alpha beta gamma Rmax h
global Am Km rho vol Ap T B a R v1 Cfeed_controller Cfeed_actual ev1 ev2 v1_end v2_end ev1i
    ev2i ev1e ev2e
global xinitial xinitial_d tcurrent xinit_OL totalcost


%Initialization
```

```matlab
C=0;

X=xinitial_d;

xinit=xinitial_d;

%tspan=[tcurrent:tstep:tf];

tspan=[tcurrent:tstep:(tcurrent+N*tstep)];


for i=1:(size(tspan,2)-1)


    %Step and hold control
    ui=U(:,i+1);


    %Differential equation solver
    [tsim,xsim] = ode23s(@ROsystemODEsOPT_controller,[tspan(i) tspan(i+1)],xinit,[],ui);


    %Resetting initial conditions
    xi=xsim(end,:);
    xinit=xsim(end,:)';


    %Determining pressure
    Ceff=Cfeed_controller*(a+(1-a)*((1-R)+(R*(v1-xi(1)))/(xi(2))));
    dpi=0.2641*Ceff*(T+273);
    Pi=((Ap*rho)/(Am*Km))*(v1-xi(1)-xi(2))+dpi;
    X=[X xi'];


    %Cost function
    C=C+(alpha*((Pi/Psp)-1)^2)+beta*((((v1-xi(1))/1.5)-1)^2)+(gamma*(((ui(1)/ev1e)-1)^2)
        +(((ui(2)/ev2e)-1)^2));


end



v1_end=X(1,end);
```

```
v2_end=X(2,end);


xinit_OL=xinit;
```

Constraints File for Model Predictive Control of Feed Flow Reversal ("FR_constraints2.m")

```
function [cineq,ceq] = FR_constraints2(U)


global x0 t0 tstep N tf h Psp alpha beta Rmax
global Am Km rho vol Ap T B a R v1 Cfeed_controller ev1 ev2 ev1i ev2i
    v1_end v2_end
global tcurrent uinitial lookup_table neg_count Rmax1 Rmax2


cineq=[];
ceq=[U(1,1)-uinitial(1,1);U(2,1)-uinitial(2,1)];
tspan=[tcurrent:tstep:tf];


for z=2:size(tspan,2);

   cineq(1,z)=-U(1,z); %ev1 is positive
   cineq(2,z)=-U(2,z); %ev2 is positive


   cineq(3,z)=abs((U(1,z)-U(1,z-1)))-Rmax1; %valve opening/closing rate
   cineq(4,z)=abs((U(2,z)-U(2,z-1)))-Rmax2;


%   cineq(3,z)=abs(U(1,z)-U(1,z-1))-abs(Rmax1-U(1,z-1)); %valve opening/closing rate
%   cineq(4,z)=abs(U(2,z)-U(2,z-1))-abs(Rmax2-U(2,z-1));


   cineq(5,z)=(v1_end+v2_end-v1); %mass balance
```

```
end
```

Main Code File for Energy Optimal Control ("EO_opt_m3_test.m")

```
%Energy Optimal MPC RO Operation - 1/10

% 1/10 - Updated valve curves for new 7 turn actuator

%M3 System Version (UDP connectivity)


clear all

%delete(instrfindall)


options= optimset('LargeScale','off','MaxFunEvals',20000,'MaxIter',300,'Display',
    'notify','TolFun',1e-4);


global t0 tstep N tf h v_p_set alpha beta

global Am Km rho vol Ap Cfeed fos

global xinitial tcurrent uinitial v_r_temp v_p_set v_p_temp


%----UDP Transmission Code (port opening and data collection----

% m3_LabView_receive = udp('localhost','LocalPort',50055,'RemotePort',50056);

% m3_LabView_send = udp('192.168.1.101','LocalPort',50057,'RemotePort',50058);

% fopen(m3_LabView_receive);

% fopen(m3_LabView_send);

%-------------------------------------------------------------


%--------Model Parameters---------

Am=15.6; %m^2 membrane surface area

Km=6.4e-9; %s/m membrane permeability (Km/rho=Lp)

rho=1007; %kg/m^3 water density

vol=0.6; %m^3 system volume 0.04
```

```
Ap=1.26e-4; %m^2 pipe cross-sectional area

m3s_to_gpm=15850.3231; %conversion between m3/s and gpm

fos = 78.7; %parameter for calculating osmotic pressure (pi_0 = fos * Cfeed)

%--------------------------------


%--------Manipulated Constants and Set-Points---------

tstep = 10; %seconds between receiving data from M3 LabVIEW

%-----------------------------------------------------


iter = 0;


while 1>0 %change this to some condition based on M3 settings or stop command
%
    iter = iter + 1


    %---Receive M3 states data from LabVIEW---
%     DataReceived = fscanf(m3_LabView_receive)
%     M3_states = sscanf(DataReceived,'%f') %need Cfeed, control actions
    M3_states = [3600,1,2,0.75];
    %-------------------------------------


    %---Converting feed conductivity to concentration (ppm)---
    M3_feed_conductivity = M3_states(1); %(uS)
    Cfeed = 0.4103*(M3_feed_conductivity^1.0299); %from Han's experimental fitting (NaCl)
    %---------------------------------------------------------


    %---Converting feed flow rate to velocity (m/s)---
    M3_feed_flowrate = M3_states(3); %(uS)
    v_f_current = (M3_feed_flowrate/m3s_to_gpm)/Ap;
    %-------------------------------------------------------


    %---Setting permeate flow rate---
```

```
Qp_set_gpm = M3_states(4); %gpm

Qp_set_m3s = Qp_set_gpm/m3s_to_gpm; %unit conversion

v_p_set = Qp_set_m3s/Ap;

%-------------------------------


%---Converting from valve position to valve resistance---

Op_low = 0.7;

Op_mid1 = 1.4;

Op_mid2 = 7;

Op_high = 49;


Op_M3 = M3_states(2)*7;


if Op_M3 < Op_low

    evr_current = -512287*log(Op_M3)-167284;

end

if Op_M3 < Op_mid1 && Op_M3 >= Op_low

    evr_current = -12425*log(Op_M3)+11043;

end

if Op_M3 < Op_mid2 && Op_M3 >= Op_mid1

    evr_current = -2052*log(Op_M3)+7434.3;

end

if Op_M3 < Op_high && Op_M3 >= Op_mid2

    evr_current = -1435.6*log(Op_M3)+6091.6;

end

if Op_M3 < Op_high

    evr_current = -264.83*log(Op_M3)+1554.1;

end

%----------------------------------------------------------


%---Initial control value guess---

if iter == 1;
```

```
    uinitial = [v_p_set*2; evr_current]; %using current M3 control settings

    %uinitial = [v_p_set; 2300]; %using current M3 control settings

else

    uinitial = uinitial_prev;

end

%-------------------------------


%---Initial State---

if iter == 1;

    xinitial = 1.2;

else

    xinitial = xinitial_prev;

end

%------------------


%---Nonlinear optimization code---

clear v_p_temp


uinitial

%U = fmincon('EO_costfunction_m3',uinitial,[],[],[],[],[0.4;265],[5;77500],

    'EO_constraints_m3',options);

U = fmincon('EO_costfunction_m3',uinitial,[],[],[],[],[0.5;428],[5;200000],

    'EO_constraints_m3',options);

opt_control = U %final optimal control values

%-------------------------------


%---Model integration---

[tsim,xsim] = ode23s(@EO_ROsystem_ODE_m3,[0 tstep],xinitial,[],opt_control);


xinitial_prev = xsim(end); %resetting initial state to final state of previous integration

uinitial_prev = opt_control;

%----------------------
```

```matlab
%---Conversion of valve resistance to valve position (based on
%experimental data)---


%---Case 1---
if opt_control(2) < 523

    valve_position = exp((opt_control(2)-1554.1)/-264.83);

end
%---Case 2---
if opt_control(2) >= 523 && opt_control(2) < 3352

    valve_position = exp((opt_control(2)-6091.6)/-1435.6);

end
%---Case 3---
if opt_control(2) >= 3352 && opt_control(2) < 6991

    valve_position = exp((opt_control(2)-7434.3)/-2052);

end
%---Case 4---
if opt_control(2) >= 6991 && opt_control(2) < 15566

    valve_position = exp((opt_control(2)-11043)/-12425);

end
%---Case 5---
if opt_control(2) >= 15566

    valve_position = exp((opt_control(2)+167284)/-512287);

end


%---Data formatting for transmission back to LabVIEW---
Q_f_gpm = (opt_control(1)*Ap)*m3s_to_gpm; %unit conversion from m/s to gpm


send_data = [Q_f_gpm , (valve_position/7)] %actuator operating points
%----------------------------------------------------

%---Data transmission---
```

```
%    fprintf(m3_LabView_send,'%f %f',send_data);

    %----------------------


end


% fclose(m3_LabView_receive);

% fclose(m3_LabView_send)

% delete(m3_LabView_receive);

% delete(m3_LabView_send);

% clear m3_LabView_receive

% clear m3_LabView_send
```

Constraints File (fmincon function) for Energy Optimal Control ("EO_constraints_m3.m")

```
function [cineq,ceq] = EO_constraints(U)


global Am Km rho vol Ap fos Cfeed
global v_r_temp v_f_temp SEC_temp v_p_set v_p_temp


cineq = [];


ceq(1) = [v_p_set-v_p_temp];
ceq(2) = [isreal(v_f_temp)-1];
ceq(3) = [isreal(U(2))-1];


%---Calculations for Thermodynamic Restriction Constraint---
Y = v_p_temp/v_f_temp;
dpi = fos*Cfeed*((log(1/(1-Y)))/Y);


pi_0 = fos*Cfeed;
pi_0 = pi_0/6895; %conversion from Pa to psi


Pi = ((Ap*rho)/(Am*Km))*(v_f_temp-v_r_temp)+dpi;
Pi = Pi/6895; %conversion from Pa to psi
%----------------------------------------------------------


cineq(1) = -U(1)+v_p_set; %VFD setting is positive
cineq(2) = -U(2); %Valve setting is positive
cineq(3) = -SEC_temp; %SEC is positive
cineq(4) = ((pi_0)/(1-Y))-Pi; %Thermodynamic restriction
```

Cost Function File (fmincon function) for Energy Optimal Control ("EO_costfunction_m3.m")

```
function C = EO_costfunction(U)

global h Am Km rho vol Ap fos Cfeed
global xinitial v_r_temp v_f_temp v_p_temp SEC_temp

options = optimset('Display','off');

%Differential equation solver
xsim = fsolve(@EO_ROsystem_ODE_ss_m3,xinitial,options,U);

%Resetting initial conditions
v_r_temp = xsim(end);
v_f_temp = U(1);
e_vr_temp = U(2);

%Determining recovery, pressure, SEC
v_p_temp = v_f_temp-v_r_temp
Y = v_p_temp/v_f_temp;
dpi = fos*Cfeed*((log(1/(1-Y)))/Y);
Pi = ((Ap*rho)/(Am*Km))*(v_f_temp-v_r_temp)+dpi;
SEC_temp = Pi/Y;

%Cost function
C = SEC_temp;
```

RO System ODE File (fmincon function) for Energy Optimal Control ("EO_ROsystem_ODE_m3.m")

```
function dv=EO_ROsystem_ODE(t,v_r,u)
global Am Km rho vol Ap Cfeed fos


%---Control Action---
v_f=u(1);
ev_r=u(2);
%-------------------


%---Recovery, Osmotic Pressure Calculations---
Y=(v_f-v_r)/(v_f);
dpi=fos*Cfeed*((log(1/(1-Y)))/Y);
%------------------------------------------------------------------------


%----System ODEs----
dv=(((Ap^2)/(Am*Km*vol))*(v_f-v_r)+((Ap*dpi)/(rho*vol)))-0.5*((Ap*ev_r*v_r^2)/vol);
     %retentate stream
%------------------
```

RO System Steady State Equation File (fmincon function) for Energy Optimal Control ("EO_ROsystem_ODE_ss_m3.m")

```
function dv=EO_ROsystem_ODE(v_r,u)


global Am Km rho vol Ap Cfeed fos


%---Control Action---
v_f=u(1);
ev_r=u(2);
%-------------------


%---Recovery, Osmotic Pressure Calculations---
Y=(v_f-v_r)/(v_f);
dpi=fos*Cfeed*((log(1/(1-Y)))/Y);
%----------------------------------------------------------------------


%----System ODEs----
dv=(((Ap^2)/(Am*Km*vol))*(v_f-v_r)+((Ap*dpi)/(rho*vol)))-0.5*((Ap*ev_r*v_r^2)/vol);
     %retentate stream
%-------------------
```

Stand-alone Image Analysis Program - Main File ("stand_alone_tester.m")

```
%Stand-alone image series tester for Image Analysis Paper 8/5/09

%Alex Bartman


%Directory Selection

prompt = {'Saved Camera Images Directory:'};

name = 'Import Images';

num_lines = 1;

def = 'E:\Run\fig\';

options.Resize='on';

options.WindowStyle='normal';

options.Interpreter='tex';

directory_name=uigetdir(def,'Select image directory:');

directory_list=dir([char(directory_name),'\','*.jpg'])

directory_name=[directory_name,'\'];

number_of_files=length(directory_list);

directory_name=char(directory_name);


im1path=([directory_name,directory_list(1).name])

image1=imread(im1path);

imageRef=rgb2gray(image1);

[nrows,ncols]=size(imageRef);  %obtaining image size of first image (for confirmedCrystals)

confirmedCrystals=zeros(nrows,ncols); %perm scaling image

lifeCount=zeros(nrows,ncols);


    compTol1=0.68;

    compTol2=0.71;


fraccoverp=[1,0];


numCrystals_prev=0;
```

```matlab
crystalCount = [];


% Initialization of the membrane structure

membrane.tcrystals=0; % total number of crystals

membrane.ncrystals=[]; %at each time step

membrane.naggregates=[]; %at each time step

membrane.crystals=[]; % array of crystals in the membrane

membrane.taggregates=0; % total number of crystals

membrane.aggregates=[]; % array of aggregates in the membrane


test1 = zeros(nrows,ncols);


%mov = avifile('detection3.avi','FPS',2);

% mov2 = avifile('gradient.avi','FPS',2);


for i=2:number_of_files;


    i

    number_of_files


    im1path=([directory_name,directory_list(i-1).name]);

    im2path=([directory_name,directory_list(i).name]);


    confirmedCrystals_prev=confirmedCrystals;

    lifeCount_prev=lifeCount;


    %----------Detection Function--------------
    [confirmedCrystals,lifeCount,membrane,test1,BW2]=IMcomparev2(im1path,im2path,
        confirmedCrystals,lifeCount,
      compTol1,compTol2,number_of_files,membrane,test1);


    permscale1=length(find(confirmedCrystals==1));
```

```matlab
%[movie_frame,map]=gray2ind(confirmedCrystals,2);

%mov = addframe(mov,movie_frame);

%    [movie_frame2,map]=gray2ind(test1,100);

%    mov2 = addframe(mov2,movie_frame2);




%----Crystal count-----

[labelMatrix, numCrystals] = bwlabel(BW2,8);


pixelLength = 0.0042;


pixelDist = .0192/pixelLength; % maximum distance of separation for pixels of
    %the same crystal

%bw_allLabel = bwlabel(bw_all,8);

N = max(max(labelMatrix));

% if there are any crystals and any are 3 or less pixels from

% another, combine those crystals

for j = 1:N

    currentDist = bwdist(labelMatrix==j);

    changePixels = unique(labelMatrix(currentDist<=pixelDist));

    changePixels = setdiff(changePixels,[0 j]);

    labelMatrix(ismember(labelMatrix,changePixels))=j;

end

newPixelUnique = setdiff(unique(labelMatrix),0);

for j = 1:length(newPixelUnique)

    labelMatrix(labelMatrix == newPixelUnique(j))=j;

end

N = max(max(labelMatrix));

numCrystals = N;

%% the variable bw_all is the binary image of surface coverage

%% pixelLength is the length of each pixel in cm/pixel
```

```matlab
        if numCrystals < numCrystals_prev %|| numCrystals > (numCrystals_prev +

            %0.1*numCrystals_prev)

            numCrystals = numCrystals_prev;

        end

        numCrystals_prev = numCrystals;


        crystalCount = [crystalCount;[i, numCrystals]];

        %----------------------

        fraccover2=permscale1/(nrows*ncols);



        %creating a matrix of coverage values for plotting on the screen

        fraccoverp=[fraccoverp;[i,(fraccover2*100)]];


end


%mov = close(mov);

% mov2 = close(mov2);


load('run2data.mat', 'run2actual');

load('run3actual.mat', 'run3actual');

load('run3full.mat', 'run3full');

load('run4actual.mat', 'run4actual');

load('run4full.mat', 'run4full');

load('run5actual.mat', 'run5actual');

%[r2r, r2c]=size(run2actual);


run2data = [0,0];


%run2

% for j=1:number_of_files;
```

```
%      if run2actual(j,3)~=0
%          run2data = [run2data; [j, run2actual(j,3)]];
%      end
% end
%
% figure
% plot(fraccoverp(:,1),fraccoverp(:,2),'r-');
% hold on
% plot(run2data(:,1),run2data(:,2),'b-');
% legend('Image program', 'James data');
% figure
% plot(crystalCount(:,1),crystalCount(:,2))


%run3
% run3data=[];
% for j=1:number_of_files;
%          run3data = [run3data; [j, run3actual(j,2)]];
% end
%
% figure
% plot(fraccoverp(:,1),fraccoverp(:,2),'r-');
% hold on
% plot(run3data(:,1),run3data(:,2),'b-');
% legend('Image program', 'James data');


%run3 full
% run3data=[];
% run3full=[0,0,0;run3full];
% for j=1:number_of_files;
%      if run3full(j,3)~=0
%          run3data = [run3data; [j, run3full(j,3)]];
%      end
```

```
% end
%
% figure
% plot(fraccoverp(:,1),fraccoverp(:,2),'r-');
% hold on
% plot(run3data(:,1),run3data(:,2),'b-');
% legend('Image program', 'James data');
% figure
% plot(crystalCount(:,1),crystalCount(:,2))
% hold on
% run3cc = [3;12;31;46;57;60];
% plot([0;run3data(:,1)],[0;run3cc])


%run4
% run4data=[];
% for j=1:number_of_files;
%         run4data = [run4data; [j, run4actual(j,2)]];
% end
%
% figure
% plot(fraccoverp(:,1),fraccoverp(:,2),'r-');
% hold on
% plot(run4data(:,1),run4data(:,2),'b-');
% legend('Image program', 'James data');


%run4 full
% run4data=[];
% run4full=[0,0,0;run4full];
% for j=1:number_of_files;
%     if run4full(j,3)~=0
%         run4data = [run4data; [j, run4full(j,3)]];
%     end
```

```
% end

%

% figure

% plot(fraccoverp(:,1),fraccoverp(:,2),'r-');

% hold on

% plot(run4data(:,1),run4data(:,2),'b-');

% legend('Image program', 'James data');

% figure

% plot(crystalCount(:,1),crystalCount(:,2))




%run5

run5data=[];

for j=1:number_of_files;

        run5data = [run5data; [j, run5actual(j,2)]];

end


figure

plot(fraccoverp(:,1),fraccoverp(:,2),'r-');

hold on

plot(run5data(:,1),run5data(:,2),'b-');

legend('Image program', 'James data');

figure

plot(crystalCount(:,1),crystalCount(:,2))

hold on

run5cc = [1;1;2;3;3;4;6;6;7;7;7;9;10;10;12;13;15;15;16;18;23;25;26;27;39;52;55;66;66;75];

plot([0;run5data(:,1)],[0;run5cc],'r-')
```

Stand-alone Image Analysis Program - Image Comparison Function (includes code from Robert Rallo) ("IMcomparev2.m")

```
function [confirmedCrystals,lifeCount,membrane,test1,BW2]=IMcomparev2(im1path,im2path
    ,confirmedCrystals,lifeCount,
  compTol1,compTol2,number_of_files,membrane,test1)


%reference image
%image1=imread('C:\Users\Alex\Desktop\Panoche MATLAB\4_26_16_18_250psi_220_1.33_
    %0.5ppm_\image000.jpg');
image1=imread(im1path);
%image for comparison
image2=imread(im2path);
%---------------------------------------------------------------


%Image Conversion (uint8: 0-255)
imageRef=rgb2gray(image1);
imageComp=rgb2gray(image2);


im0Abs=imabsdiff(imageComp,imageRef); %Reference image is substracted


% figure
% imshow(im0Abs)


imtest = imsubtract(imageComp,imageRef);
imtest_pos = find(imtest>0);
imtest_neg = find(imtest<0);


test1(imtest_pos) = test1(imtest_pos) + 1;
test1(imtest_neg) = test1(imtest_neg) - 1;
```

```
%----sigma value
sigma=1;
%--------------


% Preprocessing Phase #1: Edge indicator function
nsize=ceil(sigma*3)*2+1;          % This value depends on the size of the features
G=fspecial('gaussian',nsize,sigma);
im0Smooth=conv2(double(im0Abs),double(G),'same');  % smooth image by Gaussian convolution
[Ix,Iy]=gradient(double(im0Smooth));
f=Ix.^2+Iy.^2;
g=1./(1+f);
im01=imcomplement(g);


[maxtest1, maxtest2] = max(max(im01));


maxtest1;


% figure
% imshow(im01)


% Preprocessing Phase #2: Hysteresis Thresholding


%-----Tolerances!
maxT=compTol1;
minT=compTol2;
%----------------
im01=hysthresh(im01,maxT,minT);


% figure
% imshow(im01)


% Preprocessing Phase #3: Morphological transforms (binary image)
```

```matlab
im01=bwmorph(im01,'clean');

%im01=imdilate(im01, strel('disk',4));

im01=imfill(im01,8,'holes');


% Checking to find the new crystals


%----Confirmation Threshold---

confirmThreshold=2;

%---------------------------


newCrystals=imsubtract(confirmedCrystals,double(im01));

positions=find(newCrystals<0);

%---> newCrystals(positions)=1; %new canditate crystals

lifeCount(positions)=lifeCount(positions)+1;

newConfirmed=find(lifeCount>confirmThreshold);

confirmedCrystals(newConfirmed)=1;


TOTALIMAGES = number_of_files;

lifeCount(newConfirmed)=-(TOTALIMAGES+1);

% [centroids,Bb]=BLOBProcessing(confirmedCrystals);


confirmedCrystals=double(bwmorph(confirmedCrystals,'close'));

confirmedCrystals=imfill(confirmedCrystals,'holes');

confirmedCrystals=double(bwmorph(confirmedCrystals,'open'));

confirmedCrystals=double(bwmorph(confirmedCrystals,'clean'));

confirmedCrystals=double(bwmorph(confirmedCrystals,'bridge'));


% figure

% imshow(confirmedCrystals)


L = bwlabel(confirmedCrystals,8);

stats = regionprops(L, 'Area');
```

```
idx = find([stats.Area] > 150); %350 default for paper

BW2 = ismember(L, idx);
```

"hysthresh.m" also used [48].

Bibliography

[1] OLI Analyzer 2.0. *OLI Systems*, Morris Plains, NJ, USA, 2005.

[2] LabVIEW 2010. *National Instruments*, Austin, TX, USA, 2010.

[3] A. Abbas. Model predictive control of a reverse osmosis desalination unit. *Desalination*, 194:268–280, 2006.

[4] A. Abbas and N. Al-Bastaki. Modeling of an RO water desalination unit using neural networks. *Chemical Engineering Journal*, 114:139–143, 2005.

[5] M. Aboabboud and S. Elmasallati. Potable water production from seawater by the reverse osmosis technique in Libya. *Desalination*, 203:119–133, 2007.

[6] M. Al-haj Ali, A. Ajbar, and K. Alhumaizi. Robust model-based control of a tubular reverse-osmosis desalination unit. *Desalination*, 255:129–136, 2010.

[7] M. Al-haj Ali, A. Ajbar, E. Ali, and K. Alhumaizi. Modeling the transient behavior of an experimental reverse osmosis tubular membrane. *Desalination*, 245:194–204, 2009.

[8] M. Alahmad. Prediction of performance of sea water reverse osmosis units. *Desalination*, 261:131–137, 2010.

[9] I. Alatiqi, H. Ettourney, and H. El-Dessouky. Process control in water desalination industry: an overview. *Desalination*, 126:15–32, 1999.

[10] I. M. Alatiqi, A. H. Ghabris, and S. Ebrahim. System identification and control of reverse osmosis desalination. *Desalination*, 75:119–140, 1989.

[11] A. Alexiadis, J. Bao, D. F. Fletcher, D. E. Wiley, and D. J. Clements. Dynamic response of a high-pressure reverse osmosis membrane simulation to time dependent disturbances. *Desalination*, 191:397–403, 2006.

[12] A. Alexiadis, D. E. Wiley, A. Vishnoi, R. H. K. Lee, D. F. Fletcher, and J. Bao. CFD modelling of reverse osmosis membrane flow and validation with experimental results. *Desalination*, 217:242–250, 2007.

[13] W. S. Ang, S. Lee, and M. Elimelech. Chemical and physical aspects of cleaning organic-fouled reverse osmosis membranes. *Journal of Membrane Science*, 272:198–210, 2006.

[14] J. Z. Assef, J. C. Watters, P. B. Deshpande, and I. M. Alatiqi. Advanced control of a reverse osmosis desalination unit. *Journal of Process Control*, 4:283–289, 1997.

[15] N. Avraham, C. Dosoretz, and R. Semiat. Osmotic backwash process in RO membranes. *Desalination*, 199:387–389, 2006.

[16] A. R. Bartman, P. D. Christofides, and Y. Cohen. Non-linear model-based control of an experimental reverse-osmosis water desalination system. *Industrial and Engineering Chemistry Research*, 48:6126–6136, 2009.

[17] A. R. Bartman, C. W. McFall, P. D. Christofides, and Y. Cohen. Model predictive control of feed flow reversal in a reverse osmosis desalination process. *Journal of Process Control*, 19:433–442, 2008.

[18] A. R. Bartman, A. Zhu, P. D. Christofides, and Y. Cohen. Minimizing energy consumption in reverse osmosis membrane desalination using optimization-based control. *Journal of Process Control*, 10:1261–1269, 2010.

[19] R. B. Bird, W. E. Stewart, and E. N. Lightfoot. Transport phenomena. In *Second Edition*, pages 203–206. John Wiley and Sons, 2002.

[20] K. Bourouni, T. Ben M'Barek, and A. Al Taee. Design and optimization of desalination reverse osmosis plants driven by renewable energies using genetic algorithms. *Renewable Energy*, 36:936–950, 2011.

[21] A. C. Burden, P. B. Deshpande, and J. C. Watters. Advanced control of a B-9 Permasep permeator desalination pilot plant. *Desalination*, 133:271–283, 2001.

[22] M. Busch and W. E. Mickols. Reducing energy consumption in seawater desalination. *Desalination*, 165:299–312, 2004.

[23] F.H. Butt, F. Rahman, and U. Baduruthamal. Identification of scale deposits through membrane autopsy. *Desalination*, 101:219–230, 1995.

[24] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8:679–698, 1986.

[25] J. Chen, F. Wang, M. Meybeck, D. He, X. Xia, and L. Zhang. Spatial and temporal analysis of water chemistry records (19582000) in the Huanghe (Yellow River) basin. *Global Biogeochemical Cycles*, 19:3016–3040, 2005.

[26] J. C. Chen, Q. Li, and M. Elimelech. In situ monitoring techniques for concentration polarization and fouling phenomena in membrane filtration. *Advances in Colloid and Interface Science*, 107:83–108, 2004.

[27] P. D. Christofides and N. H. El-Farra. *Control of Nonlinear and Hybrid Process Systems: Designs for Uncertainty, Constraints and Time-Delays,* 446 pages. Springer, New York, 2005.

[28] Y. Cohen and M. Uchymiak. Method and system for monitoring reverse osmosis membranes, patent pending. *WO 2007/087578.*

[29] N. H. El-Farra and P. D. Christofides. Integrating robustness, optimality, and constraints in control of nonlinear processes. *Chemical Engineering Science*, 56:1841–1868, 2001.

[30] N. H. El-Farra and P. D. Christofides. Bounded robust control of constrained multivariable nonlinear processes. *Chemical Engineering Science*, 58:3025–3047, 2003.

[31] N. H. El-Farra, P. Mhaskar, and P. D. Christofides. Hybrid predictive control of nonlinear systems: Method and applications to chemical processes. *International Journal of Robust and Nonlinear Control*, 14:199–225, 2004.

[32] C. J. Gabelich, T. I. Yun, B. M. Coffey, and I. H. Suffet. Pilot-scale testing of reverse osmosis using conventional treatment and microfiltration. *Desalination*, 154:207–223, 2008.

[33] A. Gambier and E. Badreddin. Application of hybrid modeling and control techniques to desalination plants. *Desalination*, 152:175–184, 2002.

[34] C. J. Geankoplis. Transport processes and separation process principles. pages 883–884. Prentice Hall PTR, 2003.

[35] A. Ghobeity and A. Mitsos. Optimal time-dependent operation of seawater reverse osmosis. *Desalination*, 263:76–88, 2010.

[36] A. M. Gilau and M. J. Small. Designing cost-effective seawater reverse osmosis system under optimal energy conditions. *Renewable Energy*, 33:617–630, 2008.

[37] M. Gloede and T. Melin. Physical aspects of membrane scaling. *Desalination*, 224:71–75, 2008.

[38] H. Gu. Demonstration of rapid membrane characterization and real-time membrane module performance analysis using an automated reverse osmosis desalination pilot plant. Masters thesis, University of California, Los Angeles, 2010.

[39] H. Gu, A. R. Bartman, M. Uchymiak, P. D. Christofides, and Y. Cohen. Demonstration of automated feed flow reversal mode of RO operation without chemical additives in a RO pilot system with real-time gypsum scale monitoring. *In preparation*, 2011.

[40] S. Hargrove and S. Ilias. Flux enhancement using flow reversal in ultrafiltration. *Separation Science and Technology*, 34:1319–1331, 1999.

[41] D. Hasson, A. Drak, and R. Semiat. Inception of CaSO4 scaling on RO membranes at various water recovery levels. *Desalination*, 139:73–81, 2001.

[42] S. G. J. Heijman, E. Rabinovitch, F. Bos, N. Olthof, and J. C. van Dijk. Sustainable seawater desalination: stand-alone small scale windmill and reverse osmosis system. *Desalination*, 248:114–117, 2009.

[43] D. Herold and A. Neskakis. A small PV-driven reverse osmosis desalination plant on the island of gran canaria. *Desalination*, 137:285–292, 2001.

[44] A. Hossam-Eldin, A. M. El-Nashar, and A. Ismaiel. Techno-economic optimization of swro desalination using advanced control approaches. *Desalination and Water Treatment*, 12:389–399, 2009.

[45] K. Jamal, M. A. Khan, and M. Kamil. Mathematical modeling of reverse osmosis systems. *Desalination*, 160:29–42, 2004.

[46] I. C. Karagiannis and P. G. Soldatos. Water desalination cost literature: review and assessment. *Desalination*, 223:448–456, 2008.

[47] M. Kim, J. Au, A. Rahardianto, J. Glater, Y. Cohen, F. W. Gerringer, and C. J. Gabelich. Impact of conventional water treatment coagulants on mineral scaling in ro desalting of brackish water. *Industrial and Engineering Chemistry Research*, 48:3126–3135, 2009.

[48] P. Kovesi. hystresh.m. *School of Computer Science & Software Engineering, University of Western Australia*, Copyright 1996-2005.

[49] M. Krstic. Performance improvement and limitations in extremum seeking control. *Systems and Control Letters*, 39:313–326, 2000.

[50] E. Kujundzic, K. Cobry, A. R. Greenberg, and M. Hernandez. Use of ultrasonic sensors for characterization of membrane fouling and cleaning. *Journal of Engineered Fibers and Fabrics*, 3:35–44, 2008.

[51] D. A. Ladner, A. Subramani, M. Kumar, S. S. Adham, and M. M. Clark. Bench-scale evaluation of seawater desalination by reverse osmosis. *Desalination*, 250:490–499, 2010.

[52] Y. G. Lee, Y. S. Lee, J. J. Jeon, S. Lee, D. R. Yang, I. S. Kim, and J. H. Kim. Artificial neural network model for optimizing operation of a seawater reverse osmosis desalination plant. *Desalination*, 247:180–189, 2009.

[53] R.A. Leishear. Dynamic pipe stresses during water hammer: A finite element approach. *Journal of Pressure Vessel Technology*, 129:226–233, 2007.

[54] C. C. K. Liu, J. Park, R. Migita, and G. Qin. Experiments of a prototype wind-driven reverse osmosis desalination system with feedback control. *Desalination*, 150:277–287, 2002.

[55] C. C. K. Liu, W. Xia, and J. W. Park. A wind-driven reverse osmosis system for aquaculture wastewater reuse and nutrient recovery. *Desalination*, 202:24–30, 2007.

[56] Y. Lu, Y. Hu, X. Zhang, L. Wu, and Q. Liu. Optimum design of reverse osmosis system under different feed concentration and product specification. *Journal of Membrane Science*, 287:219–229, 2007.

[57] E. Lyster, J. Au, R. Rallo, F. Giralt, and Y. Cohen. Coupled 3-d hydrodynamics and mass transfer analysis of mineral scaling-induced flux decline in a laboratory plate-and-frame reverse osmosis membrane module. *Journal of Membrane Science*, 339:39–48, 2009.

[58] E. Lyster and Y. Cohen. Numerical study of concentration polarization in a rectangular reverse osmosis membrane channel: Permeate flux variation and hydrodynamic end effects. *Journal of Membrane Science*, 303:140–153, 2007.

[59] A. P. Mairal, A. R. Greenberg, W. B. Krantz, and L. J. Bond. Real-time measurement of inorganic fouling of RO desalination membranes using ultrasonic time-domain reflectometry. *Journal of Membrane Science*, 159:185–196, 1999.

[60] L. Malaeb and G. M. Ayoub. Reverse osmosis technology for water treatment: state of the art review. *Desalination*, 267:1–8, 2011.

[61] T. Manth, M. Gabor, and E. Oklejas. Minimizing RO energy consumption under variable conditions of operation. *Desalination*, 157:9–21, 2003.

[62] M. G. Marcovecchio, P. A. Aguirre, and N. J. Scenna. Global optimal design of reverse osmosis networks for seawater desalination: modeling and algorithm. *Desalination*, 184:259–271, 2005.

[63] C. W. McFall, A. R. Bartman, P. D. Christofides, and Y. Cohen. Control and monitoring of a high recovery reverse osmosis desalination process. *Industrial and Engineering Chemistry Research*, 47:6698–6710, 2008.

[64] P. Mhaskar, N. H. El-Farra, and P. D. Christofides. A framework for classical controller tuning using nonlinear control techniques. *AIChE Journal*, 51:3292–3299, 2005.

[65] P. Mhaskar, N. H. El-Farra, and P. D. Christofides. Predictive control of switched nonlinear systems with scheduled mode transitions. *IEEE Transactions on Automatic Control*, 50:1670–1680, 2005.

[66] P. Mhaskar, N. H. El-Farra, and P. D. Christofides. Stabilization of nonlinear systems with state and control constraints using Lyapunov-based predictive control. *Systems and Control Letters*, 55:650–659, 2006.

[67] R. Y. Ning, T. L. Troyer, and R. S. Tominello. Chemical control of colloidal fouling of reverse osmosis systems. *Desalination*, 172:1–6, 2005.

[68] B. A. Ogunnaike and W. H. Ray. *Process Dynamics, Modeling, and Control.* Oxford University Press, New York, 1994.

[69] C. Park, P. Park, P. P. Mane, H. Hyung, V. Gandhi, S. Kim, and J. Kim. Stochastic cost estimation approach for full-scale reverse osmosis desalination plants. *Journal of Membrane Science*, 364:52–64, 2010.

[70] R.H. Perry and D.W. Green. Perry's chemical engineers' handbook: 7th edition. pages 6–44. McGraw-Hill, 1997.

[71] N. Pomerantz, Y. Ladizhansky, E. Korin, M. Waisman, N. Daltrophe, and J. Gilron. Prevention of scaling of reverse osmosis membranes by "zeroing" the elapsed nucleation time. part I. calcium sulfate. *Industrial and Engineering Chemistry Research*, 45:2008–2016, 2006.

[72] MATLAB R2010a. *The Mathworks*, Natick, MA, USA, 2010.

[73] A. Rahardianto, J. Gao, C. J. Gabelich, M. D. Williams, and Y. Cohen. High recovery membrane desalting of low-salinity brackish water: Integration of accelerated precipitation softening with membrane RO. *Journal of Membrane Science*, 289:123–137, 2007.

[74] A. Rahardianto, B. C. McCool, and Y. Cohen. Reverse osmosis desalting of inland brackish water of high gypsum scaling propensity: kinetics and mitigation of membrane mineral scaling. *Environmental Science and Technology*, 42:4292–4297, 2008.

[75] A. Rahardianto, W. Shih, R. Lee, and Y. Cohen. Diagnostic characterization of gypsum scale formation and control in RO membrane desalination of brackish water. *Journal of Membrane Science*, 279:655–668, 2006.

[76] G. Raluy, L. Serra, and J. Uche. Life cycle assessment of MSF, MED and RO desalination technologies. *Energy*, 31:2361–2372, 2006.

[77] J. B. Riggs and M. N. Karim. *Chemical and Bio-Process Control, 3rd ed.* Ferret, Lubbock, Texas, 2006.

[78] Spirax Sarco. Steam and condensate loop book. In *ISBN 978-0-9550691-4-7*. Spirax Sarco, 2007.

[79] S. Skogestad. Plantwide control: the search for the self-optimizing control structure. *Journal of Process Control*, 10:487–507, 2000.

[80] J. M. Smith, H. C. Van Ness, and M. M. Abbott. Chemical engineering thermodynamics. In *Sixth Edition*, pages 600–603. McGraw-Hill, 2001.

[81] L. Souari and M. Hassairi. Sea water desalination by reverse osmosis: the true needs for energy. *Desalination*, 206:465–473, 2007.

[82] K. S. Spiegler and O. Kedem. Thermodynamics of hyperfiltration (reverse osmosis): criteria for efficient membranes. *Desalination*, 1:311–326, 1966.

[83] M. Uchymiak. *Optimization of RO desalting using external RO monitoring.* Ph.D. thesis, University of California, Los Angeles, 2010.

[84] M. Uchymiak, A. R. Bartman, N. Daltrophe, M. Weissman, J. Gilron, P. D. Christofides, W. J. Kaiser, and Y. Cohen. Brackish water reverse osmosis (BWRO) operation in feed flow reversal mode using an ex situ scale observation detector (EXSOD). *Journal of Membrane Science*, 341:60–66, 2009.

[85] M. Uchymiak, E. Lyster, J. Glater, and Y. Cohen. Kinetics of gypsum crystal growth on a reverse osmosis membrane. *Journal of Membrane Science*, 314:163–172, 2008.

[86] M. Uchymiak, A. Rahardianto, E. Lyster, J. Glater, and Y. Cohen. A novel RO ex situ scale observation detector (EXSOD) for mineral scale characterization and early detection. *Journal of Membrane Science*, 291:86–95, 2007.

[87] C. A. C. van de Lisdonk, J. A. M. van Paassen, and J. C. Schippers. Monitoring scaling in nanofiltration and reverse osmosis membrane systems. *Desalination*, 132:101–108, 2000.

[88] W. G. J. van der Meer and J. C. van Dijk. Theoretical optimization of spiral-wound and capillary nanofiltration modules. *Desalination*, 113:129–146, 1997.

[89] A. Villafafila and I. M. Mujtaba. Fresh water by reverse osmosis based desalination: simulation and optimisation. *Desalination*, 155:1–13, 2003.

[90] J. G. Wijmans and R. W. Baker. The solution-diffusion model: a review. *Journal of Membrane Science*, 107:1–21, 1995.

[91] M. Wilf. Design consequences of recent improvements in membrane performance. *Desalination*, 113:157–163, 1997.

[92] M. Wilf and C. Bartels. Optimization of seawater RO systems design. *Desalination*, 173:1–12, 2005.

[93] L. Wurth, R. Hannemann, and W. Marquardt. Neighboring-extremal updates for nonlinear model-predictive control and dynamic real-time optimization. *Journal of Process Control*, 19:1277–1288, 2009.

[94] Z. Zhang, V. M. Bright, and A. R. Greenberg. Use of capacitive microsensors and ultrasonic time-domain reflectometry for in-situ quantification of concentration polarization and membrane fouling in pressure-driven membrane filtration. *Sensors and Actuators B*, 117:323–331, 2006.

[95] A. Zhu, P. D. Christofides, and Y. Cohen. Effect of thermodynamic restriction on energy cost optimization of RO membrane water desalination. *Industrial and Engineering Chemistry Research*, 48:6010–6021, 2009.

[96] A. Zhu, P. D. Christofides, and Y. Cohen. Energy consumption optimization of reverse osmosis membrane water desalination subject to feed salinity fluctuation. *Industrial and Engineering Chemistry Research*, 48:9581–9589, 2009.

[97] A. Zhu, P. D. Christofides, and Y. Cohen. Minimization of energy consumption for a two-pass membrane desalination: effect of energy recovery, membrane rejection and retentate recycling. *Journal of Membrane Science*, 339:126–137, 2009.

[98] A. Zhu, P. D. Christofides, and Y. Cohen. On RO membrane and energy costs and associated incentives for future enhancements of membrane permeability. *Journal of Membrane Science*, 344:1–5, 2009.

[99] A. Zhu, A. Rahardianto, P. D. Christofides, and Y. Cohen. Reverse osmosis desalination with high permeability membranes - cost optimization and research needs. *Desalination and Water Treatment*, 15:256–266, 2010.

[100] M. Zhu, M. M. El-Halwagi, and M. Al-Ahmad. Optimal design and scheduling of flexible reverse osmosis networks. *Journal of Membrane Science*, 129:161–174, 1997.

[101] A. Zilouchian and M. Jafar. Automation and process control of reverse osmosis plants using soft computing methodologies. *Desalination*, 135:51–59, 2001.