

Real-Time Optimization of Interconnected Systems via Modifier Adaptation, with Application to Gas-Compressor Stations

THÈSE N° 8666 (2018)

PRÉSENTÉE LE 21 SEPTEMBRE 2018

À LA FACULTÉ DES SCIENCES ET TECHNIQUES DE L'INGÉNIEUR

LABORATOIRE D'AUTOMATIQUE - COMMUN

PROGRAMME DOCTORAL EN ROBOTIQUE, CONTRÔLE ET SYSTÈMES INTELLIGENTS

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Predrag MILOSAVLJEVIC

acceptée sur proposition du jury:

Prof. C. N. Jones, président du jury
Prof. D. Bonvin, Dr T. Faulwasser, directeurs de thèse
Prof. N. Thornhill, rapporteuse
Dr M. Mercangöz, rapporteur
Prof. D. Kuhn, rapporteur



ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2018

Acknowledgements

This thesis represents part of my work as a PhD student at the Laboratoire d'Automatique at École Polytechnique Fédérale de Lausanne. Undertaking it has been a truly valuable and life-changing experience for me and it would not have been possible to do it without the support and guidance I received from numerous people who have contributed to creating a productive and pleasant work. Therefore, I wish to express my warmest gratitude to all those persons whose comments, suggestions, criticism, support and encouragement, both personal and academic, have had a considerable bearing on this work.

First and foremost, I would like to thank my advisor Prof. Dominique Bonvin for his continued support over the years and to express my sincere gratitude for allowing me to grow as a research scientist. In countless number of times he gave me confidence, encouragement and steered me in the right direction whenever he thought I needed it. I am thankful to him for making LA a fantastic place for sharing exciting research ideas. During my studies, I had the pleasure to collaborate with many post-docs in the lab. Thus, I thank my thesis co-director Dr. Timm Faulwasser, who has contributed to improve the overall quality of my work. I have learned a lot from very lively and productive discussions with him. Along the same line, I thank Dr. Alejandro Marchetti for his sharp and insightful feedback on certain segments of my work and for transmitting to me his vision about the field. Also, I am grateful to Dr. Rene Schneider for being a great source of encouragement in the last years of my doctoral studies, for his interest in my research ideas and creating an excellent team-work atmosphere. It was very enjoyable working with him within as well as outside the LA. Special appreciation also goes to Tafarel and Martand, with whom I shared many interesting discussions related to RTO and everything that surrounds it! Without them, this journey would have been much more difficult. I also thank Andrea Cortinovis and Dr. Mehmet Mercangöz for their insightful comments and feedback on my work regarding compressors. Their willingness to discuss scientific ideas was of great help to me. I would like to thank Prof. Nina Tornhill, Prof. Daniel Kuhn and Dr. Mehmet Mercangöz, and Prof. Colin Jones for accepting the invitation to be part of my thesis committee and for accepting to evaluate my thesis and its defense.

Moreover, I thank all the other professors at the Laboratoire d'Automatique, including Prof. Roland Longchamp, Prof. Colin Jones, Dr. Alireza Karimi, and Prof. Giancarlo Ferrari Trecate, for fostering a great working atmosphere in the lab. I would like to thank Dr. Alireza Karimi for all fruitful discussions, for his patience and for tirelessly explaining and re-explaining the control theory. My sincere appreciation also goes to Christophe for infinite patience and help. I thank the secretaries, Ruth, Eva, Francine, Margot and Nicole for making

Acknowledgements

my life easier and always helping with great patience and professionalism. Thanks to Sandra and Phillipe for the help with students and for always being there for a friendly chat.

This professional adventure at LA has also enabled me to enrich my personal life and make great and long-lasting friendships. I thank all my labmates for a positive atmosphere and for making all difficult moments less stressful and more enjoyable. I am happy to have met Diogo, Shriniketh, Sohail, Ivan, Ehsan, Altug, Christoph, Georgios, Mahdieh, Sean, Milan, Luca, Harsh, Ioannis, Ye, Jean-Hubert, Francisco, Sanket, Faran, Petr, Michele, Pulkit, Mustafa and David. A special mention goes to Tafarel, Martand, Tomasz and Andrea with whom I shared so many enjoyable moments and memorable trips that I will always appreciate!

Now, big thanks to all my Serbian friends who have been part of my life over the past 5 years. I thank Nenad Bascarević and Kosta Jovanović, my friends, colleagues and beyond that, for believing in me, their constant support and all unforgettable moments. Also, I am grateful to Prof. Veljko Potkonjak in front of the whole ETF Robotics team!

A big thank to family Ranković for their support at the very beginning of my stay in Switzerland. Special thanks to my friends Lenka, Igor, Nataša i Zlatko, Tamara, Sloba, Mara i Baki, Klara, Edi i Milenko, Miško, Andrej, Mica i Miloš, Nevena i Laza, Boki i Brana, Jelena i Ilija for all birthdays, parties, weddings, travels, hikes, BBQs, skiing and for making my life in Switzerland just amazing!

Želim da se zahvalim svojim roditeljima na bezuslovnoj podršci i ljubavi koju su mi pružili. I wish to thank my loving parents for providing me with unfailing support and continuous encouragement in all my life pursuits. My deepest thanks and appreciation goes to my beloved mother Radoslava and father Radivoje, brother Nenad, my little nephew Voin and niece Nera, for being the greatest source of my happiness. This accomplishment would not have been possible without them.

And finally, the biggest thanks and deepest gratitude goes to my beloved wife Ana. Not only for your endless support and love, but also for your encouragement and your infinite patience! Thank you for constantly reminding me of how life is beautiful.

Lausanne, May 2018

P. M.

Abstract

The process industries are characterized by a large number of continuously operating plants, for which optimal operation is of economic and ecological importance. Many industrial systems can be regarded as an arrangement of several subsystems, where outputs of certain subsystems are inputs to others. This gives rise to the notion of interconnected systems. Plant optimality is difficult to achieve when the model used in optimization is inaccurate or in the presence of process disturbances. However, in the presence of plant-model mismatch, optimal operation can be enforced via specific real-time optimization methods. Specifically, this thesis considers so-called Modifier-Adaptation schemes which achieve plant optimality by direct incorporation of process measurements in the form of first-order corrections.

As a first contribution, this thesis proposes a novel problem formulation for modifier adaptation. Specifically, it is focused on plants consisting of multiple interconnected subsystems that allows problem decomposition and application of distributed optimization strategies. The underlying key idea is the use of measurements and global plant gradients in place of an interconnection model.

As a second contribution, this thesis investigates modifier adaptation for interconnected systems relying on local gradients by using an interconnection model. We show that the use of local information in terms of model, gradients and measurements is sufficient to optimize the steady-state performance of the plant.

Finally, we propose a distributed modifier-adaptation algorithm that, besides the interconnection model and local gradients, employs a coordinator. For this scheme, we prove feasible-side convergence to the plant optimum, where a coordinator ensures that the local optimal inputs computed for each subsystem are consistent with the interconnection model.

The experimental effort necessary to estimate the plant gradients increases with the number of plant inputs and may become intractable and sometimes not feasible or reliable for large-scale interconnected systems. The proposed approaches that use the interconnection model and local gradients overcome this problem.

As an application case study of industrial relevance, this thesis investigates the problem of optimal load-sharing for serial and parallel gas compressors. The aim of load-sharing optimization is operating compressor units in an energy-efficient way, while at the same time satisfying varying load demands. We show how the structure of both the parallel and serial compressor configurations can be exploited in the design of tailored modifier adaptation algorithms based on efficient estimation of local gradients. Our findings show that the complexity of this estimation is independent of the number of compressors. In addition, we dis-

Acknowledgements

cuss gradient estimation for the case where the compressors are operating close to the surge conditions, which induces discontinuities in the problem.

Key words: Real-Time Optimization, Uncertain Systems, Interconnected Systems, Modifier Adaptation, Gas Compressors, Optimal Load Sharing.

Résumé

Les industries de type « process » sont caractérisés par un grand nombre de procédés qui fonctionnent continuellement et pour lesquels le fonctionnement optimal a une importance économique. La composition de procédés multiples, où les sorties de certains sont les entrées des autres, s'appelle système interconnecté. L'optimalité de procédé est difficile à atteindre si le modèle utilisé pour l'optimisation est imprécis ou en présence de perturbations. Le fonctionnement optimal peut être assuré par les méthodes d'optimisation en temps réel (Real Time Optimization). Cette thèse étudie la méthode « Modifier Adaptation » qui corrige ces déficiences par l'intégration directe des mesures de procédé dans le problème d'optimisation via les valeurs de contraintes et les estimations du gradient du procédé.

Dans cette thèse on analyse la formulation du problème de procédés comprenant de multiples sous-systèmes et on décompose le problème d'optimisation pour que la stratégie décentralisée d'optimisation puisse être appliquée. Les méthodes 'Modifier Adaptation' qui utilisent les mesures de procédés et les gradients globaux en place du modèle interconnecté sont considérées. On montre que les schémas proposés optimisent la performance en régime stationnaire.

De plus, on considère le 'Modifier Adaptation' pour les systèmes interconnectés nécessitant des estimations locales du gradient. Comme l'algorithme utilise la connaissance du modèle des interconnexions, on montre que l'usage de l'information locale du modèle, les gradients et les mesures suffisent à optimiser la performance en régime stationnaire. Ensuite, on propose un algorithme de « distributed modifier adaptation » qui utilise un coordinateur en combinaison avec le modèle des interconnexions, et on démontre qu'il converge depuis le côté faisable vers l'optimum du procédé. Le coordinateur assure que les entrées locales optimales calculées par chaque sous-système sont consistantes avec le modèle des interconnexions.

L'effort expérimental nécessaire à estimer les gradients des procédés augmente avec le nombre des entrées de procédés. Il peut devenir difficile de résoudre, voire impossible, ou pas fiable pour les grands systèmes interconnectés. La méthode proposée dans cette thèse corrige ces déficiences par l'intégration des mesures de procédé locales uniquement dans le problème d'optimisation.

Également, le problème de la partition optimale de charge pour les compresseurs de gaz connectés en série et en parallèle est examiné dans cette thèse. Le but d'optimisation de la partition de charge est le fonctionnement de ces compresseurs avec consommation minimale, tout en satisfaisant différentes demandes. La 'Modifier Adaptation' qui utilise les gra-

Acknowledgements

dients locaux est appliqué au problème de la partition optimale de charge pour les compresseurs de gaz connectés en série et en parallèle. On suppose que le nombre de compresseurs actifs est constant. On montre comment la structure de l'ensemble des compresseurs connectés en parallèle ou en série peut être exploitée dans le but d'estimer les gradients locaux. De plus, nos découvertes montrent que la complexité de cette estimation est indépendante de nombre de compresseurs. Le schéma proposé marche en coopération avec le régulateur du niveau de base qui doit assurer le suivi du débit massique total dans le cas de compresseurs en parallèle, ou d'une pression de refoulement dans le cas de compresseurs en série. Également, on discute l'estimation de gradients dans le cas où les compresseurs fonctionnent près des conditions de pompage. L'activation de régulateurs anti-pompage présente un défi important pour Modifier Adaptation parce qu'il introduit une discontinuité dans les gradients à estimer. Les simulations démontrent l'efficacité de la méthode d'optimisation en temps réel présentée ici.

Mots clefs : Optimisation en temps réel, Systèmes incertains, Systèmes interconnectés, Modifier Adaptation, Compresseurs de gaz, Partition optimale de charge.

Contents

Acknowledgements	iii
Abstract (English/Français)	v
List of figures	xi
List of tables	xiv
1 Introduction	1
1.1 Motivation	1
1.1.1 Real-Time Optimization	1
1.1.2 Optimal Operation of Compressors	4
1.2 State of the Art	7
1.2.1 Real-Time Optimization	7
1.2.2 Optimization of Gas Compressors	12
1.3 Thesis Organization and Contributions	15
2 Preliminaries	19
2.1 Static Optimization	19
2.2 Modifier Adaptation	20
2.2.1 Standard Modifier Adaptation	20
2.2.2 Modifier Adaptation with Convex Upper Bounds	23
2.2.3 Gradient Estimation	26
2.2.4 Bounds on Gradient Uncertainty	27
2.2.5 Computation of Gradient Modifiers	28
2.3 Operation of a Compressor	29
2.3.1 Compressor Model	29
2.3.2 Compressor Maps, Characteristics and Efficiency	31
3 Modifier Adaptation for Interconnected Systems using Global Modifiers	33
3.1 Problem Formulation	34
3.1.1 Plant Optimization Problem Including Output and Internal Variables	35
3.1.2 Standard Plant Optimization Problem	35
3.2 Centralized MA for Interconnected Systems	36
3.2.1 MA with a Model for the Output and Internal Variables	37

Contents

3.2.2	Modeling an Interconnected System	38
3.2.3	Proposed Centralized MA Scheme	40
3.3	Distributed MA without Interconnection Model	41
3.3.1	Assumptions regarding the Inequality Constraints	42
3.3.2	Distributed MA without Interconnection Model	44
3.3.3	Distributed MA without Interconnection and Output Models	46
3.4	Numerical Example	48
3.4.1	Plant Description	48
3.4.2	Available Model	49
3.4.3	Simulation Results	49
3.5	Conclusion	50
4	Modifier Adaptation for Interconnected Systems using Local Modifiers	53
4.1	Problem Formulation	54
4.1.1	Plant Optimization Problem	54
4.1.2	Model Optimization Problem	55
4.2	Centralized MA with Interconnection Model	55
4.3	Distributed MA with Interconnection Model	57
4.3.1	Distributed MA Formulation	58
4.3.2	A Feasible-Side Convergent Distributed MA Algorithm	61
4.4	Estimation of Modifiers from Past Operating Points	63
4.5	Examples	65
4.5.1	Numerical Example	65
4.5.2	Optimization of Serial Gas-Compressor Station	68
4.6	Conclusion	73
5	Application to Gas-Compressor Stations	75
5.1	Problem Formulation	77
5.1.1	Underlying Control Loops	77
5.1.2	Parallel Configuration	77
5.1.3	Serial Configuration	79
5.2	Load-Sharing Optimization via MA	80
5.2.1	MA for Parallel Compressors	80
5.2.2	MA for Serial Compressors	82
5.2.3	Implementation Aspects	83
5.3	Gradient Estimation Exploiting the Problem Structure	84
5.3.1	Direction of Perturbations	85
5.3.2	Gradient Estimation for Parallel Configuration	85
5.3.3	Gradient Estimation for Serial Configuration	86
5.3.4	Estimation of Discontinuous Compressor Function Gradients	89
5.4	Simulation Results	90
5.4.1	Parallel Configuration	91
5.4.2	Serial Configuration	92

5.5 Conclusion	99
6 Conclusions	101
6.1 Final Remarks	101
6.2 Outlook and Perspectives	103
A Modifier Adaptation with Convex Upper Bounds and Inexact Gradients	107
A.1 Numerical Example	112
B A Link Between MA and Proximal-Gradient Algorithm	115
C Proofs	117
C.1 Proof of Lemma 3.1	117
C.2 Proof of Proposition 3.1	118
C.3 Proof of Proposition 3.2	118
C.4 Proof of Proposition 4.1	119
C.5 Proof of Proposition 4.3	120
Bibliography	137
Curriculum Vitae	138

List of Figures

1.1	Real-time optimization as a part of a multilayer control structure	2
1.2	Path for natural gas from the well to the user	5
2.1	Diagram of a single compressor.	30
2.2	Compressor operating constraints and cost	31
3.1	Plant with the inputs \mathbf{u} , the outputs \mathbf{y}_p and the internal variables \mathbf{v}_p	35
3.2	Model of the plant with the inputs \mathbf{u} , the outputs \mathbf{y} and the internal variables \mathbf{v}	37
3.3	Model of a plant composed of interconnected subsystems	39
3.4	Model of the i -th subsystem.	39
3.5	Coordination-free structure of Algorithms 3.2 and 3.3.	46
3.6	Structure of the plant with two interconnected subsystems.	48
3.7	Convergence of the input sequences and the plant cost.	50
4.1	Plant composed of interconnected subsystems	55
4.2	Coordination-based structure of Algorithm 4.2.	60
4.3	Input-output structure of two interconnected subsystems.	66
4.4	Convergence of the input variables \mathbf{u} and interconnection variables \mathbf{v} via Algorithm 4.2.	67
4.5	Number of model-based inner-loop iterations of Algorithm 4.2 (top) and plant cost evolution (bottom).	67
4.6	Schematic diagram of serial compressors.	69
4.7	Compressor efficiency maps	70
4.8	Evolution of the serial compressor station inputs via distributed MA with local modifiers	72
4.9	Performance loss of the serial compressor station via distributed MA with local modifiers	72
4.10	Value of the constraint $G_2(\mathbf{z}_2)$ through RTO iterations of Algorithm 4.3.	73
5.1	Parallel and serial compressor configuration with control structure	78
5.2	Discontinuity in the gradient of the compressor cost and constraints	82
5.3	Operating steady-state points for a compressor in parallel and serial configuration	87
5.4	Discontinuity in the gradient of the compressor functions	90

List of Figures

5.5 Compressor efficiency maps for the plant and model 91

5.6 Time profiles of the normalized speeds for parallel configuration 93

5.7 Plant surge control distance for parallel configuration 94

5.8 Normalized total station mass flow and power loss for parallel configuration . . 95

5.9 Gradient error contour lines 96

5.10 Time profiles of the normalized speeds for serial configuration 97

5.11 Plant surge control distance for serial configuration 98

5.12 Normalized discharge pressure $p_{d,p,3}$ and power loss for serial configuration . . 99

A.1 RTO iterates for Algorithm 2.2 and scenarios in Table A.1 114

List of Tables

A.1 Four different scenarios corresponding to Figure A.1	113
--	-----

1 Introduction

1.1 Motivation

1.1.1 Real-Time Optimization

In a world of increasing competition and emerging energy crisis, operating systems optimally is a must. Over the last decades, the bridge between the design of a control system and optimization of a complex dynamic process has attracted considerable attention in industry, in spite of limited quality of process models and unmeasured varying disturbances affecting process operation. A well-established approach to create a link between control and performance improvement is Real-Time Optimization (RTO). An RTO system is an upper-level control system that is operated in closed loop and provides setpoints to the lower-level control systems in order to maintain the process operation as close as possible to optimality. RTO was first applied in the chemical industry more than 30 years ago. Nowadays, RTO implementations are intended for a range of applications in the chemical and petrochemical industry, biological processes, mineral production, food production processes as well as in energy systems. These processes require continuous monitoring and adjustment in order to keep them operate efficiently, while taking into account security, quality, environmental and equipment constraints. These tasks are of immense importance in today's highly competitive markets. The process, oil and gas industries are investing significantly to upgrade their operations. For such high-capacity plants, even a 1% improvement yields significant annual savings for a company. Legal regulations, intense competition, changes in the markets and requirements for more environmental-friendly applications are some of the reasons that motivate the industries to improve their current practices.

A general objective of operating industrial processes is to maximize economical efficiency of the plant. In complex systems, the straightforward approach of designing and implementing a single centralized control unit is quite difficult and in many cases of complex multi-variable processes just impossible. Rather, optimal operation of an industrial plant is typically addressed by a decision hierarchy involving several layers as shown in Figure 1.1. A well-established way to cope with such a complex system is to apply a hierarchical control



Figure 1.1 – Real-time optimization as a part of a multilayer control structure

structure which is essentially a cascade structure (Findeisen et al., 1980). The idea is to decompose the original control task into a sequence of simpler and hierarchically structured subtasks. There are two basic methods of decomposition of the overall control objective:

- (1) *multilayer (functional) decomposition*
- (2) *multilevel (spatial) decomposition.*

The first one, so-called *multilayer structure*, leads to a vertical decomposition of automation tasks. The decision unit connected with each layer makes decisions concerning the controlled process, but each of them makes decisions of a different kind. The RTO level provides the bridge between plant scheduling and process control.

In the second one, so-called *multilevel structure*, the spatial decomposition is done horizontally. It is based on the control task being distributed into the local subtasks of the same functional kind but related to individual spatially isolated parts of the entire complex control process. It results in subtasks of smaller dimensionality with smaller amount of processed information. Namely, the plant is spatially decomposed into interacting groups of processing units. The exchange of information between these interacting processing units is often done from time to time via a coordinator. Each unit of a plant is highly integrated with the overall process. It gives rise to approaches known as decentralized optimizing control (Morari et al., 1980), optimizing control of interconnected systems (Brdyś and Tatjewski, 2005) or plant-wide control (Stephanopoulos and Ng, 2000; Larsson and Skogestad, 2000). Important

reasons for using decentralized or multilevel rather than centralized control structures are: (i) the need to increase the overall system robustness so that the system still remains operational when one of the units fails or when an information link breaks down; (ii) the possibility of decreasing the system sensitivity to disturbances if local control units can respond faster and more adequately than one central unit; (iii) the developments in computer technology enabling application of powerful distributed computer control systems.

RTO relies on a process model that is used to compute the best operating conditions. One of the main challenges in design of RTO schemes comes from the fact that the models are simplified representatives of the reality and are subject to uncertainty. The current industry practice, the “two-step” approach, is to regularly estimate the parameters of a process model, and then to recompute the new operating points. However, this methodology has a number of important drawbacks. Improved RTO algorithms should address the following problems:

- **Imperfect models:** The availability of process operating data enables updating the process model for better prediction of the plant outputs. As one may expect, the performance of the RTO systems depends on the model accuracy. However, obtaining accurate models is quite difficult. If the model contains many uncertain parameters, it is difficult to ensure sufficient excitation in order to estimate these parameters, and doing so will prevent it from reaching the optimization objective. Parameter estimation is certainly useful as it improves the quality of the process model, which may then be used for offline analyses. Furthermore, the development of RTO methods that do not require frequent online parameter estimation is of considerable interest.
- **Optimal operation:** The issue of plant optimization using inaccurate models has been addressed in the literature. It is crucial to know whether the RTO algorithm is capable of reaching an optimal solution for the plant. Convergence to the optimum should be ensured even for considerable structural plant-model mismatch. In such a case, it may happen that the optimal solution obtained from the “two-step” approach does not satisfy the optimality conditions of the plant (Forbes et al., 1994).
- **Constraint satisfaction:** Guaranteeing satisfaction of *equipment constraints* (valve positions, compressor speeds, horsepower limits) and *safety constraints* (safety limits, critical temperatures and pressures) is of paramount importance. Namely, violation of these constraints can lead to severe economic consequences. In the presence of plant-model mismatch and disturbances, the model-based solution such as the “two-step” approach cannot guarantee that constraints are satisfied as they might be poorly predicted by the model.
- **Changing optimum:** *Degradation of the process* results in performance decrease and increase in power consumption for the same plant operating condition. From the start-up of a process until its shutdown for maintenance, the process goes through a continuous state of degradation, including wearing of mechanical equipment, fouling of

turbines, decaying of catalyst activities, etc. *External conditions* such as weather conditions, or any disturbance that occurs in some unmeasured external input to the plant, affect the operating condition of the process (cooling water temperatures, air cooler efficiencies and heat loss from equipment).

- **Convergence speed:** The convergence speed of an algorithm is quite important in RTO. It is defined by the number of setpoint changes required prior to reaching the proximity of the plant optimum. Even though, in any RTO application, the plant optimum may change due to process degradation, the assumption is that the disturbance variations occur slowly with respect to the plant settling time and to the time the RTO algorithm takes to converge. Thus, the RTO algorithm ensures the tracking of the plant optimum in case of low-frequency degradation and disturbance.

Note that, the classical “two-step” approach works well provided that structural plant-model mismatch is not significantly large and that sufficient excitation is provided for estimation of the uncertain model parameters. Unfortunately, such conditions are rarely met in practice.

An alternative RTO algorithm called Modifier Adaptation (MA) has been developed in recent years in response to the drawbacks of the “two-step” approach. MA has been applied to numerous systems proving its effectiveness in obtaining optimal operation of industrial systems (Marchetti et al., 2016). Even though optimality of the real process is guaranteed upon convergence, there are several practical issues remaining to be addressed regarding MA application to complex systems. Firstly, MA requires estimation of experimental plant gradients, that is the plant sensitivities to small changes in operating conditions. Secondly, a reliable MA formulation for optimization of plants with interconnected systems is required. This assumes a formulation that guarantees convergence to plant optimality as well as satisfaction of each subsystem’s constraints. As already mentioned, dividing the optimization tasks into local subtasks leads to a formulation where the plant-model mismatch can be handled locally with lower-dimensional problems. Namely, MA relies on the use of a model linking the plant inputs to performance. Due to the use of local plant measurements, this link can be estimated on the local level for each subsystem. As this decomposition is beneficial from a computational point of view, which is not the primary aim of the MA methodology, it can also be beneficial to efficient estimation of the plant gradients.

The effectiveness of the methodology developed in this thesis is illustrated considering the problem of optimization of compressors operating in different configurations. Thus, the following subsection recalls the importance of operation of compressors in various industrial processes.

1.1.2 Optimal Operation of Compressors

Many processes in industries use compressors to achieve different objectives for their operations. Compressors are mechanical machines, utilized in various applications such as gas

transport through a pipe, air compression for utilities, and can also recirculate fluids. In most cases, operation of compressors in industrial processes is quite energy consuming and it also imposes relatively high compressor maintenance costs.

To that end, efficient scheduling, management and operation of compressors could save energy and reduce operational costs. Energy-intensive applications of compressors can be found in the chemical and natural gas industries. In this section, we focus on the use of compressors in *natural gas systems*.

Compressors in Natural Gas Transportation Systems

Natural gas is considered as an essential energy source for the future (EIA, 2017). The statistics show that most of natural gas consumption is concentrated in the industrial and electric sectors, accounting for 87% of the total world natural gas consumption, with an average growth of 1.7% and 2.0% per year, respectively, through 2050. Global projections in natural gas reserves clearly indicate that natural gas will play an increasingly important role in supporting growth in markets through 2050. In Germany, for example, the annual consumption of compressor stations amounts to 6.5TWh (Erd, 2017). The benefits of natural gas as a primary energy source are of immense importance within natural gas end-use consumption sectors such as the *residential, industrial and electric generation sectors*.

In the residential sector, natural gas is one of the primary sources of energy used for heating. Also, the industrial sector uses natural gas as the main source of energy for heating and power generation. Operations in the industrial sector, related to oil and gas transport from the source (fields) all the way to the end-users, have been discussed by Kurz and Brun (2012b).

Natural gas is provided from wells to gas processing plants which produce dry gas that is transported to the residential sector or industries. As depicted in Figure 1.2, from the oil and gas wells until its delivery to the end-users, the compression applications of natural gas take place in different stages and can be classified into three major categories (Kurz and Brun, 2012b):

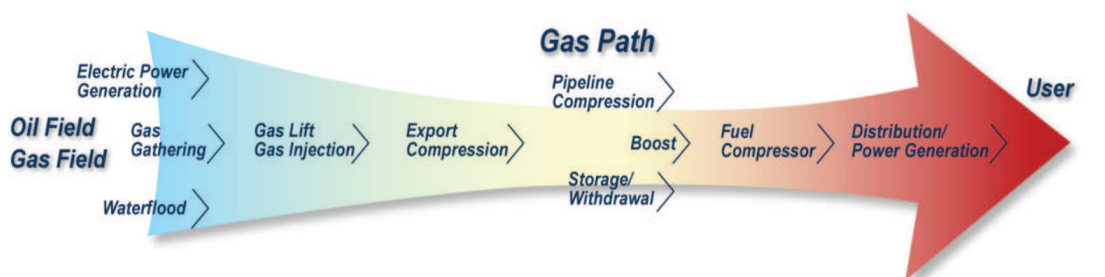


Figure 1.2 – Path for natural gas from the well to the user (Rasmussen et al., 2009).

1. **The upstream applications** are utilized for transportation of natural gas from oil and gas wells to gas processing plants:

Natural gas is extracted from the wells together with crude oil. Then, two of them are being separated in the process of gas gathering that includes the ash gas compression. At this level, compressors are used to increase the flow due to natural low pressures in wells. Gas lift and gas injection are both used to increase the flow and production from a reservoir. A major difference between the two is that the gas injection goes into the reservoir, and the gas lift goes into the well bore and to the surface. Natural gas is afterwards compressed via export gas compressor from an offshore platform through an under sea pipeline system.

2. **The midstream applications** are utilized to transport gas from the processing plant to the distribution points:

So called boost compressors are used in gas processing plants to increase gas pressure coming from the gathering system (Kurz and Brun, 2012b). Also, compressor stations are used to compress the sales gas from the exit of the plant to the delivery points. This implies gas transportation through pipelines up to 4,000 km over land and 2,000 km off-shore (Schmidt et al., 2015). In case of larger distances, the natural gas is liquefied and transported in ships. The liquefaction of gases is a complicated process that uses compression and expansions to achieve high pressures and very low temperatures. Compressors are also used to store gas in reservoirs to deal with the uncertainty in the demand of the gas.

3. **The downstream applications** consist of gas distribution to the end-users. Namely, gas distribution is the process of routing gas to individual customers. There are also compressors that increase the pressure of the gas at the exit of the plant to the pressure of the inlet of downstream pipelines.

For both transmission and distribution networks, the gas flows through various devices including pipes, regulators, valves, and compressors. In a transmission network, gas pressure is reduced due to friction with the pipe wall as the gas travels through the pipe. Some of this pressure is added back at compressor stations, which raises the pressure of the gas passing through them. This involves compressor stations in series to increase the pressure of the gas in order to overcome the friction losses in pipes. A typical example of a compressor station can involve a varying number of compressor units that operate in serial and parallel configuration. A large number of compressor stations can be found in different parts of a gas network system, for example a large network may include tens of compressor stations distributed in a strategic way together with several hundreds of pipelines (Ríos-Mercado and Borraz-Sánchez, 2015). Nevertheless, there are gas networks that involve a small number of compressor stations for gas transportation, but with great power and flow capacity. Since compressors are considered to be the most critical and most energy-consuming elements in a compressor station, minor improvements in efficiency can have a significant impact on the

operating costs. A compressor station's operating cost, however, is generally measured by the fuel/power consumed at the compressor station. According to Luongo et al. (1989), the operating cost of running the compressor stations represents between 25% and 50% of the total company's operating budget. Hence, the objective for a transmission network is to minimize the total fuel/power consumption of the compressor stations while satisfying the specified delivery flow rates and pressure requirements at the delivery terminals.

Since the problem of optimizing the gas compressor stations is of tremendous importance, it opens the door to the development of optimization algorithms. The difficulties of such optimization problems come from several aspects. First, compressor stations are very sophisticated entities themselves and they can consist of several compressor units in various configurations with different efficiencies and characteristics. The compressor behavior is nonlinear and the maintenance and activation of each compressor should be planned in the optimal way. Each station may contain several groups of compressor units of various vintages and the capacity may expand due to various demands over the years. Also, the set of constraints that define feasible operating range of the compressors together with the constraints related to the pipeline system constitute a very complex set of nonlinear constraints. Finally, operations of the valves and regulators may introduce certain discontinuities to the problems as well. These factors make compressor controller development both challenging and crucial for improving their overall efficiency.

1.2 State of the Art

1.2.1 Real-Time Optimization

RTO has emerged over the past forty years to overcome difficulties associated with plant-model mismatch. Uncertainty can be classified into three groups, namely, (i) parametric uncertainty, when the values of the model parameters do not correspond to the reality of the process at hand; (ii) structural plant-model mismatch, when the structure of the model does not correspond to process dynamics; (iii) process disturbances. Note that these three sources are not mutually exclusive. In general, there are two directions in *optimization approaches under uncertainty* that can cope with plant-model mismatch depending on the available information from the plant. Namely, if no process measurements can be used for modeling the disturbances and incorporated in the control scheme, then *robust optimization* strategies are used to cope with uncertainty. Alternatively, *measurement-based optimization* techniques incorporate process measurements in the optimization framework to combat the effect of uncertainty.

In the sequel, the measurement-based techniques will be reviewed. These methods can be classified depending on how the available measurements are used. There are basically three categories, namely, (i) methods that act at the level of the inputs also referred to as *model-free* techniques; (ii) methods that use the measurements at the level of the process

Chapter 1. Introduction

model to update model parameters; (iii) methods that use the measurements at the level of the cost and constraint functions.

RTO techniques are developed for the process industry that includes both continuous plant operating at steady state, and transient plants. Srinivasan et al. (2003) and Chachuat et al. (2009) give a comprehensive review of these RTO techniques. Also, François and Bonvin (2013b) discuss batch and semi-batch processes as examples of transient processes. These processes are usually nonlinear, multivariable and affected by process disturbances and measurement noise. Moreover, they are subject to safety constraints (e.g surge, pressures or temperatures), equipment constraints (actuator limits, compressor speeds or maximum driver power) and quality constraints (product specifications). Operation of these complex processes has been managed via online computer-aided multivariable control (Cutler and Perry, 1983; Darby et al., 2011). A large number of successful RTO applications have been reported (Marlin and Hrymak, 1997; Darby et al., 2011), and it remains a very active research field.

Model-Free Methods

In model-free RTO techniques, incorporating process measurements in the optimization framework entails directly updating the inputs in a control-inspired manner. Initially, various heuristic model-free evolutionary-search techniques were developed (Box and Draper, 1969). These techniques are based on utilizing plant measurements to find directions in the input space that improve performance of the plant. On the one hand, evolutionary techniques do not require process models, only simple calculations and, due to their simplicity, they can be implemented readily. On the other hand, these techniques are quite impractical for cases with large number of inputs and complex nonlinear behavior. A more recent model-free technique, Self-Optimizing Control (SOC) (Skogestad, 2000; Alstad and Skogestad, 2007), uses the sensitivity between the uncertain model parameters and the measured outputs to generate linear combinations of the outputs that are locally insensitive to the model parameters, and which can thus be kept constant at their nominal values to reject the effect of uncertainty. Also, Necessary Conditions of Optimality (NCO) tracking (François et al., 2005; Srinivasan and Bonvin, 2007), uses output measurements to estimate the plant NCO, which are then enforced via multivariable feedback control. Extremum-Seeking Control (ESC) is a technique where dither signals are added to the inputs so that the plant gradients are estimated online using the output measurements (Krstić and Wang, 2000). In the unconstrained case, gradient control is directly applied to drive the plant cost gradient to zero.

Methods with Model Update

The most intuitive strategy to improve plant performance is to utilize measurements to update the model. This is the main idea behind the “two-step” approach (Chen and Joseph, 1987; Marlin and Hrymak, 1997). It is the most common RTO algorithm in practice (Darby et al., 2011). Although this approach can handle arbitrarily complex systems with many in-

puts, it is fairly computationally intensive. In the “two-step” approach, measured outputs are used to update the model parameters and obtain new inputs by optimizing the updated model. This way, the model is expected to represent more accurately the plant at its current operating point. The idea is to repeat this procedure until convergence is reached under the assumption that the updated model is capable of predicting the optimal plant inputs. The requirements for this to happen are referred to as the model-adequacy conditions (Forbes et al., 1994). Despite the popularity of the two-step method, these conditions will almost never be satisfied in a practical setting. Namely, in practice, the “two-step” approach is likely to perform well if a structurally correct model is available. Nevertheless, this cannot be guaranteed. Moreover, in the presence of structural plant-model mismatch, parameter estimation may be ineffective and can even lead to worse performance than if no RTO was performed at all (Agarwal et al., 1997; Marchetti et al., 2009; Gao and Engell, 2005). Also, the “two-step” approach is unlikely to perform well if there are too many uncertain parameters in the model. Therefore, another class of model-based algorithm, which addresses the issues associated with the two-step approach, has been developed in parallel.

Fixed-Model Methods

Fixed-model methods utilize both a nominal process model and appropriate measurements for guiding the iterative scheme towards optimality. But instead of refining the parameters of a first-principles model through RTO iterations, the measurements are used to update the cost and constraint functions in the optimization problem in order to improve plant performance. Early progress was made via the method called *Integrated System Optimization and Parameter Estimation* (ISOPE), which uses measurements to update both the model parameters and the gradient of the cost function in the optimization problem to be solved online Roberts (1979). A number of researchers have improved and extended the ISOPE algorithm over the years (Roberts, 1979, 1995; Roberts and Williams, 1981) and a comprehensive review of this development is given by Brdyś and Tatjewski (2005). ISOPE requires both cost measurements and estimates of the gradients of the plant cost with respect to the inputs. Due to the utilization of these gradient modifiers, ISOPE can guarantee plant optimality in spite of the presence of plant-model mismatch. Researches utilized gradient modification due to the nature of the necessary conditions of optimality that include both constraints and sensitivity conditions (Bazaraa et al., 2006). Namely, by incorporating estimates of the plant gradients in the model, the goal is to enforce NCO matching between the model and the plant.

An important improvement of the ISOPE algorithm was introduced by Tatjewski (2002) by eliminating the parameter estimation step. Furthermore, Gao and Engell (2005) refined this simplified algorithm to address plant-model mismatch in constraints. Finally, Marchetti et al. (2009) provided a solid theoretical basis for the simplified ISOPE algorithm, which resulted in the development of a fixed-model RTO scheme called Modifier Adaptation. Over the years, MA has been successfully applied to a number of complex industrial processes such as an experimental solid-oxide fuel-cell stack (Bunin et al., 2010), a simulated oxygen-consumption

plant (Navia et al., 2012) and the simulated heat and power system of a sugar and ethanol plant (Serralunga et al., 2013). An excellent overview of MA approaches and all its varieties is given in (Marchetti et al., 2016). Essentially, MA uses measurements of plant constraints and estimates of plant gradients to modify the cost and constraint functions in the model-based optimization problem without updating the model parameters. Many different MA formulations have been investigated recently with extension to closed-loop systems (Costello et al., 2013) and extension to discontinuous systems (Serralunga et al., 2014). Also, various theoretical aspects have been tackled regarding the convergence analysis of MA, such as the use of convex models to ease the numerical optimization and enforce model adequacy (François and Bonvin, 2013a), the use of second-order modifiers (Faulwasser and Bonvin, 2014), and even promising preliminary results on sufficient conditions for global convergence (Bunin, 2014; Marchetti et al., 2017; Faulwasser and Bonvin, 2014). Besides this, the gradient estimation problem has been discussed by Bunin et al. (2013); Marchetti (2013); Rodger and Chachuat (2010); Navia et al. (2013). Recently, a variant of MA known as directional modifier adaptation has been proposed (Costello et al., 2016). It uses the available process model to identify a small number of critical input directions that are crucial for plant performance. Consequently, it often suffices to estimate the plant gradients in these directions, thereby making the overall approach less expensive.

Optimization of Interconnected Systems

In the literature, interconnected systems with interacting subsystems are sometimes called *large-scale systems* (Brdys et al., 1990), *networked systems* (Venkat et al., 2008), *multi-agent systems* (Olfati-Saber et al., 2007), *partitioned systems* (Farina et al., 2010), or simply *interconnected systems* (Sandell et al., 1978; Brdyś and Tatjewski, 2005). These terms emphasize the interactions between the subsystems, which are typically caused by an exchange of material, energy or information.

Steady-state optimizing control is often performed in a two-layer hierarchical framework, where the values of the setpoints are computed in an upper supervisory control layer and transmitted down to a lower regulatory control layer. Optimizing control techniques and algorithms are available which treat the process under control as consisting of a set of interconnected subprocesses, each with its own regulatory control system and supervisory decision unit. Coordination is then required to take account of the interconnections between the subprocesses. In this structure, the supervisory control layer is itself structured into two levels. The lower level consists of a set of separate decision units whose tasks are to compute the optimal setpoints to be applied to their local regulatory controllers. The upper level consists of a single decision unit that should provide the coordination function. Such a hierarchical structure has a particular utility within distributed systems controlling a large-scale plant consisting of geographically separated subprocesses.

The computation of setpoints is usually performed by solving the steady-state optimizing control problem of minimizing, or maximizing a performance index subject to a mathemati-

cal model, together with constraints representing the steady-state behavior of the controlled process. Inevitably, the model will be an approximation, in structure and parameters, to reality and thus the solution will in general be suboptimal. In order to take into account the plant-model differences, various techniques that use real process measurements have been developed over the years.

An early strategy is known as the *Interaction Balance Method with Feedback* (IBMF) (Finden et al., 1980). While it has been observed to yield significant performance improvement, it does not necessarily lead to plant optimality. Also, it requires a large number of online iterations and thus lacks practical value. Gu and Wan (2001) proposed a method to deal with these issues. The main idea behind this method is that the model coefficients of each subprocess are replaced by corresponding fuzzy numbers. To overcome the limitation of suboptimality of the IBMF, researchers have tried to complement this distributed algorithm with elements of centralized ISOPE techniques. Similarly to the centralized case in ISOPE, plant-model mismatch can be handled by using process measurements to iteratively modify the gradient of the cost function of the optimization problem and update the parameters of the steady-state model. As a result, the plant optimum can be reached upon convergence despite plant-model mismatch (Brdys and Tatjewski, 2005). By combining these ideas with the distributed structure of IBMF, various distributed ISOPE techniques have been obtained. An overview of these techniques is given by Brdys and Tatjewski (2005). In fact, these approaches are hierarchical in nature, and the solution algorithms have a nested iterative structure.

There are several variants depending on the types of measurements and decision variables:

- The measurements for each subsystem can include only output or both inputs and outputs (Lin et al., 1989).
- The decision variables applied to the plant can include only the setpoints computed in the outer loop (model-based algorithms) or those computed in both the outer and inner loops (system-based algorithms) (Brdys et al., 1990). In the former case, a model is used for the inner iteration, and only the setpoints computed in the outer iterations are applied to the plant, thereby reducing the number of time-consuming setpoint changes (Bryds et al., 1989). Also, Lin et al. (1988) have allocated more attention to the issue of reducing the number of controller setpoint changes than to the convergence features. A variable augmentation technique is introduced to reduce the sensitivity of the algorithm to iterative gain selection and to improve the convergence of the algorithm (Lin et al., 1989; Brdys et al., 1987)

It is very important to point out that the literature on *RTO algorithms for interconnected systems* puts emphasis on handling the plant-model mismatch. Thus, we focus on the formulations for optimization of plants with interconnected systems, posed and treated as an RTO problem.

None of the approaches, referenced above, tackles the issue of gradient estimation of complex plants, rather the assumption of exact plant gradient estimates is imposed. With all approaches, convergence to the plant optimum in the presence of plant-model mismatch is possible, but the plant inequality constraints must be known exactly. Note as well that the model update is based on “two-step” approach techniques that assume frequent estimation of model parameters. Unfortunately, for complex industrial plants, accurate estimation of many parameters is extremely complicated to implement. Also, all subprocesses need to exchange model, measurements and gradient information among themselves in order to ensure the convergence of the algorithm.

This thesis presents RTO methodologies to deal with the optimal operation of interconnected systems and discusses their application to the problem of optimal distribution of the load in the compressor plants. Thus, the following subsection presents the literature review on the topic of optimal operation of compressor stations.

1.2.2 Optimization of Gas Compressors

The topic of the optimization of compressor stations has been explored by numerous researchers. Compressors are used in various applications where the nature of each application influences the objectives and constraints of the optimization problem. Also, the utilization of compressors is reported to be energy-intensive in many industrial processes (Saidur et al., 2010; Mahmoudimehr and Sanaye, 2013). The steady-state optimal operation of gas compressors in different process systems has been explored by many researchers (Han et al., 2004; Hasan et al., 2009; Sun and Ding, 2014). Scheduling of compressors is discussed by van den Heever and Grossmann (2003). Other studies focused on the optimal control of compressors using model predictive control strategies (Cortinovis et al., 2015; Gopalakrishnan and Biegler, 2013; Zavala, 2014)

Over the years, compressors have been employed in a wide range of industrial applications, particularly for gas transportation and storage in natural gas networks, extraction, processing and utilities (de Marco et al., 2011; Borraz-Sánchez and Haugland, 2013; Ríos-Mercado and Borraz-Sánchez, 2015).

Compressor stations, typically composed of several compressor units connected in series or in parallel, play a crucial role in the natural gas industry. A compressor unit is a device used to increase the pressure of natural gas by reducing its volume, thus providing the required propel force or boost to keep it moving along the line. Compressor stations are strategically installed along gas transmission lines as important assets to the gas transport industry so as to provide enough energy to natural gas for its transmission. An excellent overview of the entire pipeline network optimization is given in Ríos-Mercado and Borraz-Sánchez (2015). In the literature, various approaches for the optimization of compression networks can be found starting as early as the mid 19th century, where Hax (1967) investigated the optimization of natural gas transmission. In late 70s, Edgar et al. (1978) focused on the optimal design

of natural gas transmission networks while Marqués and Morari (1988) showed first promising results on pipeline optimization in the late 80s. Compressor station optimization was considered, for example by Murphy et al. (1989); Osiadacz (1980) and Wright et al. (1998) in the late 80s and 90s. Due to the complexity of these problems, a compressor station is modeled as a single compressor unit, even if it consists of multiple compressors.

Process systems that include utilities are applications in which compressors consume most of the energy (Han et al., 2004). The analysis of these systems does not focus on the detailed modeling of the pipes since friction losses are not important compared to natural gas networks, where the length of the pipes extends to hundreds of kilometers. The optimization of parallel compressors in refrigeration systems is studied by Widell and Eikevik (2010). Namely, they report that compressors in refrigeration systems can be used in oil and gas industry for cooling a compressed gas, liquefaction of natural gas and removing liquid in a gas plant. As reported by Kurz and Brun (2012b), compressors are also the major consumers of energy in these applications.

Application that deals with the optimization of compressors in gas storage has been studied by Kurz and Brun (2010); Camponogara et al. (2012); Kurz and Brun (2012b); Silva and Camponogara (2014).

Optimal Load Sharing

Several compressors are used when the capacity of a single station is not enough to satisfy the demands on mass flow or pressure of the downstream applications.

Load-sharing optimization can be considered as an equivalent of real-time optimization used in the process industries. Also, in the context of compressor station operation, it is an extension of what is generally referred to as load-sharing control (Staroselsky and Mirsky, 1987). Parallel centrifugal compressors are used in applications that request high mass flow rates and low pressure ratios (Boyce, 2003). Serial compressor configurations are employed to increase the total discharge pressure compared to the pressure a single compressor can achieve. The compressor station receives a general target from the dispatch center, typically in the form of an hourly flow or pressure setpoint for parallel or serial configuration, respectively. This setpoint is realized via station process controller by allocating a load distribution to each compressor. In turn, individual compressor control systems peruse their local targets by adjusting the operating speeds. Load-sharing optimization extends this task of load-sharing control by considering individual compressor performance maps and optimizing specific objective such as minimizing the power consumption of all machines. Kurz et al. (2012) commented that if compressors have identical compressor maps, then the load can be equally split or they can operate at the same surge margin. In general, compressors in a station with multiple units have, however, different performance maps and efficiencies. Several authors and practitioners reported on this note (Abbaspour et al., 2005; Lipták, 2013). Furthermore, plant maintenance results in the change of compressor maps of the same com-

pressor before and after the procedure (Paparella et al., 2013; Ciccioiti et al., 2014). In addition, compressor characteristics and efficiencies change over time due to fouling and erosion (Kurz and Brun, 2012a). According to Tirnovan et al. (2008), these characteristics can be estimated and updated by using process measurements and surrogate models.

The aim of this thesis is to formulate an optimization problem to distribute the load among compressors. The optimization of fuel consumption together with load-sharing optimization problem of the parallel gas compressor plants driven by gas turbines has been discussed by Abbaspour et al. (2005); Han et al. (2004). However, practical aspects such as update of the maps and implementation of actual optimization have not been encompassed in these papers. Cortinovis et al. (2016); Xenos et al. (2015) presented an optimization framework that updates the model parameters of the compressors online. Therein, the problem of optimal load-sharing of parallel compressor stations is discussed. Optimization of serial compressor configuration has been tackled by Kumar and Cortinovis (2017).

Maintenance and Optimal Selection of Compressors

Even though the focus of this thesis is not on active selection and maintenance of compressors, a number of papers discuss this issue in the operation of gas networks. The examples of these works are mentioned by Ríos-Mercado and Borraz-Sánchez (2015); Mohamadi-Baghmolaei et al. (2014).

Nguyen et al. (2008) consider the problem of active compressor selection in natural gas pipeline operations. This selection deals with the choice of the number of operating compressors while no optimal setpoints for compressor speeds are computed. Abbaspour et al. (2005) presented a detailed mathematical model of compressor networks, where decision variables are reconstituted by the compressor steady-state speeds and the objective function to be minimized is the total fuel consumption. Furthermore, Moritz (2007) considers a mixed-integer linear programming approach for the transient optimization of compression networks. Xenos et al. (2014) and Hawryluk et al. (2010) consider optimization with multiple objectives related to compressor stations. Namely, in Xenos et al. (2014), several different objectives in connection with both operations and maintenance are combined. Hawryluk et al. (2010) carried out a formal multi-objective optimization with the aim to minimize energy consumption and maximize station throughput. Paparella et al. (2013); Cortinovis et al. (2016) studied the optimization of a compressor station by including integer variables to represent the start up and shut down costs within parallel compressor plant. The work of Mahlke et al. (2010) used a mixed-integer approach to include discrete events into the optimization of a transient problem to deal with the optimal operation of serial compressors. It is worth noting that, in all of the above-listed studies, except for Cortinovis et al. (2016) and Xenos et al. (2015), the load-sharing optimization problem is not posed and treated as an RTO problem in the context of an automation system.

Compressors in process gas applications mainly suffer from fouling. Fouling is an im-

portant mechanism leading to performance deterioration in gas turbines over time due to the adherence of particles to airfoils and annulus surfaces. A consequence of the fouling is increased power consumption and reduced efficiency of the compressor for the same load compared to a non-fouled compressor. Rao and Naikan (2008) studied the optimal washing schedule of a single compressor. Martín-Aragón and Valdés (2014) and Sánchez et al. (2009) studied the optimal scheduling of offline washing of gas turbine compressors with thermodynamic methods. The framework presented by Xenos et al. (2016) includes the basic operational constraints of the compressors considering operational aspects, such as the prediction of power consumption depending on the operational conditions, the extra power consumption due to degradation, and minimum running and minimum shut down times of the compressors.

1.3 Thesis Organization and Contributions

The objectives of this thesis are dedicated to the study of the fixed-model RTO methods. It considers the optimizing control of a plant composed of multiple subsystems or units that are physically interconnected, such that the outputs of one system influence the inputs of other subsystems. In that context, the objectives are twofold:

- This thesis analyzes the overall plant problem formulation and decomposes the optimization problem so that a decentralized optimization strategy can be applied. Also, the role of measurements of the interconnection variables is discussed, which leads to the reformulation of the optimization problem. Such reformulation and utilization of local measurements lead to the adaptation of existing fixed-model RTO methods, so that the algorithms presented still guarantee plant optimality.
- Moreover, the local adaptation of the optimization problem of subsystems is tested on two examples of gas compressor stations. Practical aspects and effectiveness of the algorithms are illustrated on a load-sharing optimization case study for parallel and serial gas compressor configuration.

The main scientific contributions of this thesis include:

- The MA framework is extended for the first time to interconnected systems. Two novel distributed algorithms using plant measurements and global modifiers in place of an interconnection model are proposed. Due to the absence of an interconnection model, no coordinator is required. Both schemes enable optimizing the steady-state performance of an interconnected plant.
- Two novel MA algorithms for interconnected systems that use the knowledge of interconnection model are proposed. Also, both schemes rely on the use of local modifiers

that are introduced for the first time in the RTO framework. We show that, upon convergence, they deliver optimal steady-state performance for the overall plant. The first scheme is centralized. The second one is a distributed MA scheme for which plant monotonic cost decrease and feasibility guarantees are provided.

- A solution to the load-sharing optimization problem for parallel and serial compressor plants for the case when the compressors may also operate on the surge control line is presented. Also, an efficient approach for obtaining accurate gradient estimates of the plant cost and constraints in the presence of noise and plant-model mismatch is proposed. Furthermore, we show that the complexity of this estimation is independent of the number of compressors in both, parallel and serial configuration.

The thesis structure is as follows:

Chapter 2: Preliminaries. The static RTO optimization problem is formulated as a nonlinear program. The effect of plant-model mismatch is discussed, which gives a good motivation for introduction of the methodology called modifier adaptation. Model adequacy is reviewed and the basics of MA are presented together with gradient estimation techniques within the RTO framework. We also discuss a recent study of MA with convex upper bounds. This chapter concludes with a description of a compressor model.

Chapter 3: MA for Interconnected Systems using Global Modifiers. In this chapter, two distributed real-time optimization algorithms based on the MA framework are proposed. In contrast to ISOPE, MA handles the uncertain cost function and the uncertain inequality constraints without parameter estimation (Marchetti et al., 2009, 2016). Nevertheless, MA is able to drive the plant to optimality. Here, such features are leveraged for the first time for interconnected systems. The two proposed MA algorithms use plant measurements in place of an interconnection model. Uncertain inequality constraints can also be handled. It is shown for both schemes that, upon convergence, the computed inputs optimize the steady-state performance of the uncertain interconnected plant. The results herein presented have been published by Schneider et al. (2017).

Chapter 4: MA for Interconnected Systems using Local Modifiers. This chapter extends the MA framework for interconnected systems to a scheme that uses only local input measurements of each subsystem for plant gradient estimates. Two MA algorithms are discussed. The first one is a centralized scheme. The second scheme is a coordinator-based distributed MA scheme that requires estimation of only local plant gradients. The proposed algorithm has a two-layer structure. In the inner loop, a coordinator ensures that the inputs of the local subproblems are consistent with the interconnection model. Upon convergence of the inner loop, these consistent inputs are applied to the plant in the outer loop. Note that the proposed scheme does not require that the models underlying each subproblem are shared with other subsystems for the purpose of obtaining the plant gradient estimates. This is possible due to a novel way of updating the modifiers within RTO framework. Furthermore, mono-

tonic cost decrease and feasibility guarantees are given for the MA scheme using a coordinator and measured interconnection variables. The results herein presented have been published by Milosavljevic et al. (2017, 2018c).

Chapter 5: Application to Gas-Compressor Stations. This chapter investigates the use of MA for load-sharing optimization in gas compression stations consisting of several compressors in both parallel and serial configurations. The centralized version of MA for interconnected systems using local modifiers, presented in Chapter 4, is applied to both configurations. The results show that optimal operation of the plant can be obtained after a few RTO iterations without having to update the model parameters or the compressor maps. An interesting feature that makes MA particularly well suited for this problem is that it is possible to excite all compressor speeds simultaneously without significantly perturbing the station mass-flow or pressure setpoint. Also, an efficient approach for obtaining accurate gradient estimates of the plant cost and constraints in the presence of noise and plant-model mismatch is proposed. These estimates are obtained locally for each subsystem. The analysis shows that the algorithm is capable of quickly converging to the plant optimum by using only local subsystem derivatives. In fact, each subsystem relies on the estimation of the local power consumption and constraint derivatives with respect to its own inputs. The presented results have been documented by Milosavljevic et al. (2016, 2018b,a).

Chapter 6: Conclusions.

This chapter concludes the thesis and discusses the perspectives in terms of new research topics.

2 Preliminaries

RTO methods are related to either steady-state or dynamic plant optimization. In the case of former, the RTO scheme aims to find the optimal steady-state values for the plant inputs, which often correspond to setpoints for lower-level controllers. The main contribution of this thesis is related to the RTO methodology for interconnected systems. Since, each subsystem can be analyzed as a single continuous process, we will introduce the RTO methodology of centralized plants. The variations and reformulations of centralized optimization problems will be used later on for the analysis of interconnected systems.

Nonlinear programming provides a framework for characterizing optimal operating conditions of a continuous process, and the properties of an optimal solution are described in Section 2.1. If a process model is available, numerical optimization can be used to approximately compute the optimal operating conditions. It is an approximate solution because the model never perfectly matches the real process. The MA algorithm, which compensates for this mismatch using measurements, is described in Section 2.2. Finally, a model to describe the operation of a single compressor is outlined in Section 2.3.

2.1 Static Optimization

The problem of finding optimal steady-state operating conditions for a continuous process is typically expressed mathematically as

$$\begin{aligned} \mathbf{u}_p^* = \underset{\mathbf{u}}{\operatorname{argmin}} \quad & \phi_p(\mathbf{u}) \\ \text{s.t.} \quad & \mathbf{g}_p(\mathbf{u}) \leq 0, \end{aligned} \tag{2.1}$$

where \mathbf{u} is the n_u -dimensional vector of decision (or input) variables; $\phi_p, \mathbf{g}_{p,j} : \mathbb{R}^{n_u} \mapsto \mathbb{R}, j = 1, \dots, n_g$, denote the cost and constraints functions, where n_g is the number of process constraints. The notation $(\cdot)_p$ is used to denote variables associated with the "plant".

Local minima of Problem (2.1) can be characterized via the necessary conditions of opti-

mality (NCO) (Bazaraa et al., 2006).

Theorem 2.1 (KKT Necessary Conditions). *Let \mathbf{u}_p^* be a (local) optimum of Problem 2.1, and assume that \mathbf{u}_p^* is a regular point of the constraints, that is, the active constraints are linearly independent. Then, there exist unique values for the n_g -dimensional vector of Lagrange multipliers, $\boldsymbol{\mu}$, such that the following first-order Karush-Kuhn-Tucker (KKT) conditions hold at \mathbf{u}_p^* :*

$$\mathbf{g}_p(\mathbf{u}_p^*) \leq \mathbf{0}, \quad (2.2a)$$

$$\boldsymbol{\mu}^\top \mathbf{g}_p(\mathbf{u}_p^*) = 0, \quad (2.2b)$$

$$\boldsymbol{\mu} \geq \mathbf{0}, \quad (2.2c)$$

$$\nabla_{\mathbf{u}} \mathbf{L}_p(\mathbf{u}_p^*) = \mathbf{0}, \quad (2.2d)$$

with the Lagrangian function defined as $\mathbf{L}_p(\mathbf{u}_p) = \phi_p(\mathbf{u}_p) + \boldsymbol{\mu}^\top \mathbf{g}_p(\mathbf{u}_p)$.

Proof. See, for example, (Bazaraa et al., 2006, Thm 4.2.13). □

The necessary conditions of optimality in (2.2) are referred to as the primal feasibility, complementary slackness, dual feasibility and stationarity conditions, respectively. These conditions must hold at any local minimum that is also a regular point of the constraints. Note that these conditions might be satisfied by a point that is not a local minimum, as they are not sufficient conditions of optimality.

2.2 Modifier Adaptation

2.2.1 Standard Modifier Adaptation

In any practical application, the steady-state input-output mappings $\phi_p(\cdot)$ and $\mathbf{g}_p(\cdot)$ in (2.1) are typically unknown, and only an approximate nonlinear steady-state models $\phi(\cdot)$ and $\mathbf{g}(\cdot)$ are available. Thus, the model-based optimization problem becomes

$$\begin{aligned} \mathbf{u}^* = \underset{\mathbf{u}}{\operatorname{argmin}} \quad & \phi(\mathbf{u}) \\ \text{s.t.} \quad & \mathbf{g}(\mathbf{u}) \leq \mathbf{0}, \end{aligned} \quad (2.3)$$

Thus, in the presence of plant-model mismatch, the model solution \mathbf{u}^* does not generally coincide with the plant optimum \mathbf{u}_p^* in (2.1).

Hence, Problem (2.1) cannot be solved directly. One approach to deal with this problem is the method of modifier adaptation (Marchetti et al., 2009). In this section, we first review its key property of dealing with uncertainty in cost and constraints. In the presence of plant-model mismatch, it has been proposed to solve Problem (2.1) iteratively via the MA approach

given as

$$\mathbf{u}_{k+1}^* = \underset{\mathbf{u}}{\operatorname{argmin}} \phi(\mathbf{u}) + (\boldsymbol{\lambda}_k^\phi)^\top (\mathbf{u} - \mathbf{u}_k) \quad (2.4a)$$

$$\text{s.t. } \mathbf{g}(\mathbf{u}) + \boldsymbol{\epsilon}_k^g + (\boldsymbol{\lambda}_k^g)^\top (\mathbf{u} - \mathbf{u}_k) \leq \mathbf{0}, \quad (2.4b)$$

where $\phi(\cdot)$ is a model of the plant cost $\phi_p(\cdot)$, $\mathbf{g}(\cdot)$ is a model of the plant constraints $\mathbf{g}_p(\cdot)$; $\boldsymbol{\epsilon}_k^g$ is the n_g -dimensional vector of zeroth-order constraint modifiers, $\boldsymbol{\lambda}_k^\phi$ is the n_u -dimensional vector of first-order cost modifiers and $\boldsymbol{\lambda}_k^g$ is the $(n_u \times n_g)$ matrix of first-order constraint modifiers. These modifiers are given as:

$$(\boldsymbol{\lambda}_k^\phi)^\top = \frac{\partial \phi_p}{\partial \mathbf{u}}(\mathbf{u}_k) - \frac{\partial \phi}{\partial \mathbf{u}}(\mathbf{u}_k), \quad (2.5a)$$

$$\boldsymbol{\epsilon}_k^g = \mathbf{g}_p(\mathbf{u}_k) - \mathbf{g}(\mathbf{u}_k), \quad (2.5b)$$

$$(\boldsymbol{\lambda}_k^g)^\top = \frac{\partial \mathbf{g}_p}{\partial \mathbf{u}}(\mathbf{u}_k) - \frac{\partial \mathbf{g}}{\partial \mathbf{u}}(\mathbf{u}_k). \quad (2.5c)$$

Moreover, we assume that the minimum \mathbf{u}_{k+1}^* exists at every iteration k and that the inputs are filtered using a constant and nonsingular gain matrix $\mathbf{K} \in \mathbb{R}^{n_u \times n_u}$ as

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \mathbf{K}(\mathbf{u}_{k+1}^* - \mathbf{u}_k) \quad (2.6)$$

before being applied to the plant. The process measurements are used to iteratively modify the model-based problem (2.4) in such a way that, upon convergence, the necessary conditions of optimality for the modified problem match those for the plant-based problem (2.1). This is made possible by using modifiers that, at each iteration, are computed as the differences between the measured and predicted values of the constraints and the measured and predicted cost and constraint gradients. This forces the cost and constraints in the model-based optimization problem to locally match those of the plant. In its simplest form, the algorithm proceeds as follows:

Algorithm 2.1 : Modifier Adaptation (Marchetti et al., 2009)

Initialization: choose the diagonal filter matrix \mathbf{K} with eigenvalues in the interval $(0, 1]$ and choose a feasible input vector \mathbf{u}_0 .

for $k = 0 \rightarrow \infty$ **do**:

1. Apply the inputs \mathbf{u}_k to the plant and wait for steady state.
2. Update the modifiers according to (2.5a)-(2.5c).
3. Solve the modified model-based optimization problem (2.4)
4. Filter the inputs according to (2.6).

end

An important property of the modifier-adaptation approach is provided by the following theorem:

Theorem 2.2 (KKT matching for Algorithm 2.1, Marchetti et al. (2009)). *Let the MA Algorithm 2.1 converge, with $\mathbf{u}_\infty = \lim_{k \rightarrow \infty} \mathbf{u}_k$. Then, \mathbf{u}_∞ is a KKT point for the plant optimization problem (2.1).*

Note that, while the KKT-matching property is a very desirable property for an RTO algorithm, it remains a theoretical result. In a real application, due to noisy measurements and inexact gradients, the algorithm will reach a neighborhood of the plant optimum.

Note also that Theorem 2.2 guaranties that, if MA converges, it will do so to the KKT point of the plant. Whether an RTO algorithm is capable of converging to the plant optimum has been initially discussed by Forbes and Marlin (1996). In RTO, this is referred to as the “Model-Adequacy” question.

Theorem 2.3 (Model Adequacy). *Let \mathbf{u}_p^* be the unique plant optimum, which is assumed to be a regular point for the n_g^a active constraints. The process model is adequate for use in MA if the reduced Hessian of the cost function ϕ is positive definite at \mathbf{u}_p^* :*

$$\mathbf{Z}^\top (\nabla^2 \phi) \mathbf{Z} > \mathbf{0}, \quad (2.7)$$

where the columns of $\mathbf{Z} \in \mathbb{R}^{n_u \times (n_u - n_g^a)}$ are a set of basis vectors for the null space of the Jacobian of the active constraints in the model-based optimization problem (2.4).

Proof. See Marchetti et al. (2009). □

Hence, in the case of MA, model adequacy is dictated by the second-order derivatives because any mismatch in the cost and constraints is corrected by the MA scheme up to the first-order derivatives. Clearly, the positive-definiteness requirement is independent of the modifier values themselves.

Theorem 2.3 implies that the use of strictly convex models in MA schemes automatically satisfies the model adequacy condition. Indeed, François and Bonvin (2013a) proposed a method to enforce the Model-Adequacy Condition for a general nonlinear model cost function by using convex approximations.

Note that, according to Theorem 2.2, if MA converges, it will do so to a plant KKT point. Next, assuming the model-adequacy criterion is met according to Theorem 2.3, MA will converge to the plant optimum. The question related to the MA convergence properties has been tackled recently by several authors (Marchetti et al., 2009, 2017; Bunin, 2014; Faulwasser and Bonvin, 2014). Here, both necessary and sufficient conditions have been proposed for the MA convergence. Namely, Bunin (2014) discussed the equivalence between MA and the

trust-region framework and exploited this relation to propose a globally convergent modifier-adaptation algorithm using already developed trust-region theory for unconstrained optimization problems. Also, Marchetti et al. (2017) presented a feasible-side globally convergent MA formulation, wherein the cost and constraint functions belong to a certain class of convex upper-bounding functions. Faulwasser and Bonvin (2014) discuss how second-order updates in MA can lead to SQP schemes. Furthermore, Milosavljevic et al. (2018c) show connection between the MA framework and proximal-gradient methods, which is discussed in Appendix B.

From a practical point of view, it is very difficult to enforce the conditions proposed in these papers. Therefore, more focus is required on practical aspects of MA such as gradient estimation, especially for the applications with high-dimensional input space. Initially, Marchetti et al. (2009) analyzed convergence of the standard MA Algorithm 2.1 for variations of the exponential filter matrix \mathbf{K} , with positive eigenvalues in the interval $(0, 1]$. Larger eigenvalues encourage more rapid convergence, but may also cause oscillating behavior, or a failure to converge at all. Smaller eigenvalues result in the MA algorithm taking more cautious steps, making convergence more likely, but at a slower pace. Also, Navia et al. (2013) and Gao et al. (2016) proposed methods that completely avoid the gradient estimation step. Instead, the cost and constraint gradient modifiers λ_k^ϕ and λ_k^g are determined at each iteration by an unconstrained gradient-free optimization routine. While this conveniently avoids gradient estimation, its drawback is that the gradient-free optimization algorithm must optimize the plant using many RTO iterations to converge to the plant optimum. Costello et al. (2016) proposed a variant of MA known as Directional Modifier Adaptation (DMA). Here, the gradients are corrected only in the subspace spanned by the most critical (i.e., sensitive) directions for a small parametric mismatch. Next, Singhal et al. (2017) extended DMA concept by computing a set of critical directions that are robust to large parametric perturbations. The core idea in DMA is that confidence in model structure can be exploited for efficient gradient estimation i.e updating the model in directions that are key for improving the plant performance.

The proposed methodology for interconnected systems in this theses goes along the line of exploiting the knowledge of the interconnection model of the plant. Namely, we discuss how this knowledge can be used for efficient gradient estimation so that with small number of setpoint changes, MA converges to the optimum of the plant consisting of interconnected systems.

2.2.2 Modifier Adaptation with Convex Upper Bounds

Marchetti et al. (2017) presented a feasible-side globally convergent RTO formulation, where the cost and constraint functions are constructed as convex upper-bounding functions. They propose to construct the required upper-bounding functions by adding quadratic terms to the modified cost and constraint functions used in standard MA. The main feature of this method is an MA algorithm guaranteeing global feasible-side convergence to a KKT point of

the plant assuming perfect plant gradient estimates.

In general, the design of an RTO algorithm should enforce the following desirable properties:

- (i) **Plant optimality:** Despite structural plant-model mismatch, a KKT point of Problem (2.1) is reached upon convergence of (2.4).
- (ii) **Plant feasibility:** All RTO iterates \mathbf{u}_k satisfy the constraints of Problem (2.1).
- (iii) **Monotonic cost improvement:** The performance is required to improve between consecutive RTO iterates.

Besides these important basic properties, one would like to have sufficiently fast convergence and sufficient robustness with respect to gradient errors.

The key to the MA scheme proposed in Marchetti et al. (2017) is the concept of convex upper-bounding functions. These are defined next.

Definition 2.1 (Convex upper-bounding function). *Let the function $f : \mathbb{R}^{n_u} \mapsto \mathbb{R}$ be continuously differentiable. Then, the convex differentiable function $f_k^U : \mathbb{R}^{n_u} \mapsto \mathbb{R}$*

$$f_k^U(\mathbf{u}) = f(\mathbf{u}_k) + (\nabla f(\mathbf{u}_k))^\top (\mathbf{u} - \mathbf{u}_k) + \frac{\delta^f}{2} \|\mathbf{u} - \mathbf{u}_k\|_2^2,$$

is said to be a quadratic convex upper-bounding function of $f(\mathbf{u})$ at \mathbf{u}_k with an upper-bounding coefficient $\delta^f \geq 0$, if it satisfies the following conditions for all $\mathbf{u}_k \in \mathbb{R}^{n_u}$:

$$f_k^U(\mathbf{u}_k) = f(\mathbf{u}_k) \tag{2.8a}$$

$$\nabla f_k^U(\mathbf{u}_k) = \nabla f(\mathbf{u}_k) \tag{2.8b}$$

$$f_k^U(\mathbf{u}) \geq f(\mathbf{u}). \tag{2.8c}$$

Upper-bounding functions can be used to enforce monotonic cost improvement in MA schemes (Marchetti et al., 2017). In particular, at every RTO iteration k , the following update is proposed:

$$\mathbf{u}_{k+1} = \underset{\mathbf{u}}{\operatorname{argmin}} \phi_k^U(\mathbf{u}) := \phi(\mathbf{u}) + \varepsilon_k^\phi + (\lambda_k^\phi)^\top (\mathbf{u} - \mathbf{u}_k) + \frac{\delta^\phi}{2} \|\mathbf{u} - \mathbf{u}_k\|_2^2 \tag{2.9a}$$

$$\text{s.t. } g_{j,k}^U(\mathbf{u}) := g_j(\mathbf{u}) + \varepsilon_k^{g_j} + (\lambda_k^{g_j})^\top (\mathbf{u} - \mathbf{u}_k) + \frac{\delta^{g_j}}{2} \|\mathbf{u} - \mathbf{u}_k\|_2^2 \leq 0, \quad j = 1, \dots, n_g, \tag{2.9b}$$

where $\varepsilon_k^{g_j} \in \mathbb{R}$ is the constraint value modifiers, $\lambda_k^\phi, \lambda_k^{g_j} \in \mathbb{R}^{n_u}$ are the cost and constraint gradient modifiers, respectively, defined in (2.5a)–(2.5c). Also, the cost value modifiers is defined as $\varepsilon_k^\phi = \phi_p(\mathbf{u}_k) - \phi(\mathbf{u}_k)$; Note that ε_k^ϕ is a constant, thus it is not included in formulation of Problem 2.4. These modifiers ensure matching function values and first derivatives of the

plant (2.1) and the modified model (2.9) at the current operating point \mathbf{u}_k . Moreover, the upper-bounding coefficients $\delta^\phi, \delta^{g_j}$ in (2.9) need to be selected such that $\phi_k^U(\mathbf{u})$ and $g_{j,k}^U(\mathbf{u})$ are convex upper-bounding functions of $\phi_p(\mathbf{u})$ and $g_{p,j}(\mathbf{u})$, respectively.

The following algorithm explains the implementation of modifier adaptation with convex upper bounds.

Algorithm 2.2 : MA with convex upper-bounding functions (Marchetti et al., 2017)

1. *Initialization*: Provide the initial point, \mathbf{u}_0 . Set $k := 0$.
 2. *Plant experiment*: Apply the set of inputs \mathbf{u}_k to the plant and wait for steady state.
 3. *Modifier computation*: Compute the modifiers as per (2.5a)–(2.5c).
 4. *New input calculation*: Compute \mathbf{u}_{k+1} by solving Problem (2.9).
 5. *Iterate*: Set $k := k + 1$ and return to Step 2.
-

The further developments are based on the technical assumptions introduced next.

Assumption 2.1 (Plant properties).

The plant optimization problem (2.1), satisfies the following conditions:

1. *We assume that the solution is a regular KKT point i.e. linear independence constraint qualification holds.*
2. *The feasible set is a nonempty compact.*
3. *$\phi_p(\mathbf{u})$ and $g_{p,j}(\mathbf{u})$ are twice continuously differentiable functions for all \mathbf{u} .*

□

Assumption 2.2 (Model properties).

$\phi(\mathbf{u})$ and $g_j(\mathbf{u})$ are twice continuously differentiable functions for all \mathbf{u} .

□

Assumption 2.3 (Exact gradient estimates).

The constrained values and the cost and constraint gradients of the plant are perfectly known at each RTO iteration.

□

The following theorem states that the RTO Algorithm 2.2 guarantees feasible-side global convergence to a KKT point of the plant.

Theorem 2.4 (Global convergence (Marchetti et al., 2017)). *Let Assumptions 2.1-2.3 hold. Then, for any feasible initial point \mathbf{u}_0 , the input sequence generated by Algorithm 2.2 converges to a KKT point of Problem (2.1) with the following properties:*

- a) All RTO iterates satisfy the plant constraints.
- b) The plant cost decreases monotonically at each iteration.

Proof. The proof is given in (Marchetti et al., 2017, Thm.1). This result is based on the idea of (Marks and Wright, 1978). \square

2.2.3 Gradient Estimation

The most challenging part of MA is the estimation of the plant gradients or the first-order modifiers. This is the most difficult aspect of the RTO methodology since the gradients cannot be measured directly and, furthermore, measurement noise is almost invariably present. This section discusses different ways of estimating gradients, since they can be obtained in many different ways (Bunin et al., 2013; Mansour and Ellis, 2003; Zhang and Forbes, 2006; Srinivasan et al., 2011). Here, the discussion is restricted to the methods associated with MA, which is why special emphasis will be placed on the methods classified for steady-state perturbations that use only steady-state data (Marchetti et al., 2010; Brdyś and Tatjewski, 1994; Marchetti, 2013). For each change in the input variables, one must wait until the plant has reached steady state before taking measurements, which can make these methods expensive since each new point requires a new plant setpoint change. Here, we will focus only on *finite-difference approximation* methods. This is the most common approach and it requires at least $n_u + 1$ steady-state operating points to estimate the gradients.

The first alternative is forward finite-difference approximation by perturbing the current RTO point. This rather straightforward approach involved perturbing each input separately around the current operating point to get an estimate of the corresponding gradient element. For instance, the i -th element of $\nabla\phi_p(\mathbf{u}_k)$, is estimated as:

$$\frac{\partial\phi_p}{\partial\mathbf{u}_i}(\mathbf{u}_k) = \frac{\tilde{\phi}_p(\mathbf{u}_k + \delta\mathbf{u}_i) - \tilde{\phi}_p(\mathbf{u}_k)}{\|\delta\mathbf{u}_i\|}, \quad (2.10)$$

where $\delta\mathbf{u}_i$ is a vector aligned with the i -th input direction. The superscript $\tilde{(\cdot)}$ denotes a noisy measurement. The same procedure is used for estimating the constraint gradients. This approach requires n_u perturbations $\delta\mathbf{u}_i$ to be carried out at each RTO iteration, and for each perturbation a new steady state must be attained.

The second alternative is finite-difference approximation using past RTO points. It consists of computing the gradients solely from measurements of the current and previously visited RTO points $\{\mathbf{u}_k, \mathbf{u}_{k-1}, \dots, \mathbf{u}_{k-n_u+1}\}$. At the k -th RTO iteration, the following matrix can be constructed:

$$\mathbf{U}(\mathbf{u}) = [\mathbf{u} - \mathbf{u}_k, \dots, \mathbf{u} - \mathbf{u}_{k-n_u+1}]^\top \in \mathbb{R}^{n_u \times n_u}. \quad (2.11)$$

Assuming the cost measurements ϕ_p are available at each iteration, we construct the follow-

ing vectors at the k -th RTO iteration:

$$\delta\tilde{\phi}_p(\mathbf{u}) = [\tilde{\phi}_p(\mathbf{u}) - \tilde{\phi}_{p,k}, \dots, \tilde{\phi}_p(\mathbf{u}) - \tilde{\phi}_{p,k-n_u+1}]^\top \in \mathbb{R}^{n_u}. \quad (2.12)$$

If $\mathbf{U}(\mathbf{u}_k)$ is nonsingular, then the set of $n_u + 1$ RTO points $\{\mathbf{u}_{k-j}\}_{j=0}^{n_u}$ is said to be poised for linear interpolation in \mathbb{R}^{n_u} . The cost gradient at \mathbf{u}_k can then be estimated as follows:

$$\widehat{\nabla\phi}_p(\mathbf{u}_k) = (\delta\tilde{\phi}_p(\mathbf{u}_k))^\top \mathbf{U}^{-1}(\mathbf{u}_k). \quad (2.13)$$

In addition, constraint gradients can be computed in a similar way.

2.2.4 Bounds on Gradient Uncertainty

It might appear that by using previously visited RTO points, it is possible to estimate the gradients ‘for free’, that is without any additional experimental burden. In reality, the steps taken by the RTO algorithm must be taken with caution to ensure good gradient estimates. Here, we discuss estimation of the bounds on gradient estimates, which is often more challenging than obtaining the estimates themselves. In case the gradient estimates are obtained via finite-difference, Brekelmans et al. (2005) proposed a deterministic quantification of the gradient error due to the finite-difference approximation (or *truncation error*) and due to measurement noise error. The expressions obtained for the total gradient error are convex functions of the step size, from which it is easy to compute the step size that minimizes the total gradient error. Following a similar approach, Marchetti et al. (2010) analyzed the gradient error and bounds on gradient error associated with (2.13).

The gradient estimation error is defined as the difference between the estimated gradient and the true plant gradient

$$(\boldsymbol{\epsilon}^\phi(\mathbf{u}))^\top = \widehat{\nabla\phi}_p(\mathbf{u}) - \frac{\partial\phi_p}{\partial\mathbf{u}}(\mathbf{u}), \quad (2.14a)$$

Note that the row vector $\widehat{\nabla\phi}_p(\mathbf{u})$ is the estimate of the plant cost gradient. From (2.13) and using $\tilde{\phi}_p(\mathbf{u}) = \phi_p(\mathbf{u}) + \nu$, (2.14a) can be split as

$$\boldsymbol{\epsilon}^\phi(\mathbf{u}) = \boldsymbol{\epsilon}^t(\mathbf{u}) + \boldsymbol{\epsilon}^n(\mathbf{u}), \quad (2.14b)$$

where ν is measurement noise, $\boldsymbol{\epsilon}^t$ is the error due to the finite-difference approximation (or truncation) and $\boldsymbol{\epsilon}^n$ is the measurement noise error,

$$(\boldsymbol{\epsilon}^t(\mathbf{u}))^\top = [\phi_p(\mathbf{u}) - \phi_{p,k}, \dots, \phi_p(\mathbf{u}) - \phi_{p,k-n_u+1}] \mathbf{U}^{-1}(\mathbf{u}) - \frac{\partial\phi_p}{\partial\mathbf{u}}(\mathbf{u}), \quad (2.14c)$$

$$(\boldsymbol{\epsilon}^n(\mathbf{u}))^\top = [\nu - \nu_k, \dots, \nu - \nu_{k-n_u+1}] \mathbf{U}^{-1}(\mathbf{u}). \quad (2.14d)$$

Assuming that $\phi_p(\mathbf{u})$ is twice continuously differentiable with respect to \mathbf{u} , then the norm

of the gradient error due to truncation can be upper bounded as follows (Marchetti et al., 2010):

$$\|\boldsymbol{\epsilon}^t(\mathbf{u})\| \leq \mathcal{E}^t(\mathbf{u}) := \frac{d_\sigma}{2} \|[(\mathbf{u} - \mathbf{u}_k)^\top (\mathbf{u} - \mathbf{u}_k) \dots (\mathbf{u} - \mathbf{u}_{k-n_u+1})^\top (\mathbf{u} - \mathbf{u}_{k-n_u+1})] \mathbf{U}^{-1}(\mathbf{u})\|, \quad (2.15a)$$

where d_σ is an upper bound on the spectral radius of the Hessian of $\phi_p(\mathbf{u})$. Also, assuming that the noisy output $\tilde{\phi}_p(\mathbf{u})$ remains within an interval δ at steady-state operation, then the norm of the gradient error due to measurement noise can be upper bounded as follows Marchetti et al. (2010)

$$\|\boldsymbol{\epsilon}^n(\mathbf{u})\| \leq \mathcal{E}^n(\mathbf{u}) := \frac{\delta}{l_{min}(\mathbf{u})}, \quad (2.15b)$$

where $l_{min}(\mathbf{u})$ is the shortest distance between all possible pairs of complement affine subspaces that can be generated from the set of points $\mathcal{S} = \{\mathbf{u}, \mathbf{u}_k, \mathbf{u}_{k-1}, \dots, \mathbf{u}_{k-n_u+1}\}$ (see Marchetti et al. (2010) for the computation of $l_{min}(\mathbf{u})$). Roughly speaking, $l_{min}(\mathbf{u})$ is the minimum of the orthogonal distances between each individual point in the set \mathcal{S} and the hyperplane passing through the remaining points. Hence, in order to keep the noise error small, the past input points should be approximately orthogonal to each other and no two points should be too close to each other.

Thus, the overall bound on gradient error is given as sum of bounds in (2.15a) and (2.15b) as

$$\mathcal{E}^\phi(\mathbf{u}) = \mathcal{E}^t(\mathbf{u}) + \mathcal{E}^n(\mathbf{u}). \quad (2.15c)$$

2.2.5 Computation of Gradient Modifiers

The most straightforward way of computing the gradient modifiers is to directly evaluate them from the estimated gradients, according to their definition

$$(\boldsymbol{\lambda}_k^\phi)^\top = \widehat{\nabla} \phi_{p,k} - \frac{\partial \phi}{\partial \mathbf{u}}(\mathbf{u}_k), \quad (2.16a)$$

$$(\boldsymbol{\lambda}_k^{g_j})^\top = \widehat{\nabla} g_{p,j,k} - \frac{\partial g_j}{\partial \mathbf{u}}(\mathbf{u}_k), \quad (2.16b)$$

where, in principle, any of the methods mentioned in Section 2.2.3 can be used to obtain the gradient estimates $\widehat{\nabla} \phi_{p,k} = \widehat{\nabla} \phi_p(\mathbf{u}_k)$ and $\widehat{\nabla} g_{p,j,k} = \widehat{\nabla} g_{p,j}(\mathbf{u}_k), \forall j = 1, \dots, n_g$. Here, $\boldsymbol{\lambda}_k^{g_j}$ is j -th column vector of modifier $\boldsymbol{\lambda}_k^g$ in (2.4).

Instead of using a set of steady-state operating points to estimate the gradients, it is possible to use the same set to directly compute the gradient modifiers by linear interpolation or linear regression. For instance, Marchetti (2013) proposed to estimate the gradient modifiers by linear interpolation using the set of $n_u + 1$ RTO points $\{\mathbf{u}_{k-i}\}_{i=0}^{n_u}$. In addition to the plant

cost vectors $\delta\tilde{\phi}_{p,k} = \delta\tilde{\phi}_p(\mathbf{u}_k)$ in (2.12),

$$\delta\tilde{\phi}_{p,k} = [\tilde{\phi}_{p,k} - \tilde{\phi}_{p,k-1}, \dots, \tilde{\phi}_{p,k} - \tilde{\phi}_{p,k-n_u}]^\top \in \mathbb{R}^{n_u}, \quad (2.17)$$

its model counterpart can be constructed at the k -th RTO iteration

$$\delta\phi_k = [\phi_k - \phi_{k-1}, \dots, \phi_k - \phi_{k-n_u}]^\top \in \mathbb{R}^{n_u}. \quad (2.18)$$

Similarly, we define the plant and model constraint vectors for all $j = 1, \dots, n_g$:

$$\delta\tilde{g}_{p,j,k} = [\tilde{g}_{p,j,k} - \tilde{g}_{p,j,k-1}, \dots, \tilde{g}_{p,j,k} - \tilde{g}_{p,j,k-n_u}]^\top \in \mathbb{R}^{n_u} \quad (2.19)$$

$$\delta g_{j,k} = [g_{j,k} - g_{j,k-1}, \dots, g_{j,k} - g_{j,k-n_u}]^\top \in \mathbb{R}^{n_u}. \quad (2.20)$$

This leads to the following expressions for the gradient modifiers

$$(\boldsymbol{\lambda}_k^\phi)^\top = (\delta\tilde{\phi}_{p,k} - \delta\phi_k)^\top \mathbf{U}_k^{-1} \quad (2.21a)$$

$$(\boldsymbol{\lambda}_k^{g_j})^\top = (\delta\tilde{g}_{p,j,k} - \delta g_{j,k})^\top \mathbf{U}_k^{-1} \quad (2.21b)$$

where input matrix $\mathbf{U}_k = \mathbf{U}(\mathbf{u}_k)$ is defined in (2.11). Here, the sample points consist of the current and n_u most recent RTO points. However, it is also possible to include the designed perturbations in the sample set. With the gradient modifiers presented in (2.21), the modified cost and constraint functions match the corresponding measured values for the plant at the current and past operating points $\{\mathbf{u}_k, \mathbf{u}_k, \dots, \mathbf{u}_{k-n_u}\}$. Marchetti (2013) discussed that this gives a better approximation of the plant cost and constraint functions, in particular for increased distances between the points. In this case, the modified cost and constraint functions should be viewed as a higher-order correction.

2.3 Operation of a Compressor

The MA methodology developed in this thesis is tested on a load-sharing problem for compressor stations consisting of several compressors in series and in parallel. This section provides the background knowledge on a single compressor model. Hence, the description of the principles of industrial centrifugal compressor and its operational aspects are introduced here. This type of compressor is usually used in process systems and natural gas applications. This section explains the major operational aspects (characteristics, performance, control methods and maintenance) of a single compressor.

2.3.1 Compressor Model

The static compressor model used in this thesis is based on the work of Cortinovis et al. (2015). As depicted in Figure 2.1, the main element is the centrifugal compressor that is surrounded by piping and valves. The main line consists of a suction side valve (*in'*), a recycle valve line

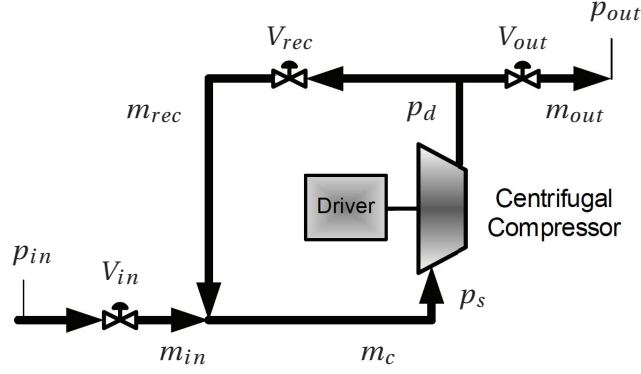


Figure 2.1 – Diagram of a single compressor.

(*'rec'*) that connects the compressor outlet with the compressor inlet, and the discharge side valve (*'out'*). The compressor conveys a gas from the upstream process to the downstream process. There is a driver that provides power to the shaft of the compressor, and it can be a *gas turbine*, *steam turbine* or an *electrical motor*. The driver rotates the shaft with rotational speed ω , which can be either constant or variable as is the case with a variable-speed drive. This type of drive is employed by a variable-speed compressor.

The dynamic compressor model can be written as:

$$\frac{dp_s}{dt} = C_1(m_{in} + m_{rec} - m_c) \quad (2.22a)$$

$$\frac{dp_d}{dt} = C_2(m_c - m_{rec} - m_{out}) \quad (2.22b)$$

$$\frac{dm_c}{dt} = C_3(p_s\Pi - p_d) \quad (2.22c)$$

$$\frac{d\omega}{dt} = C_4(\tau - \tau_{comp}) \quad (2.22d)$$

$$\frac{dm_{rec}}{dt} = C_5(m_{rec,ss} - m_{rec}), \quad (2.22e)$$

where m_c , m_{rec} , m_{in} and m_{out} denote the compressor, recycle, inlet, and outlet flows, respectively; p_s is the suction pressure, p_d the discharge pressure, Π the pressure ratio, τ the applied torque, whereas τ_{comp} is the torque resulting from the air compression, while C_i , $i = 1, \dots, 5$, are constant parameters.

Furthermore, one can write:

$$m_{in} = k_{in}V_{in}\sqrt{|p_{in} - p_s|} \quad (2.23a)$$

$$m_{out} = k_{out}V_{out}\sqrt{|p_d - p_{out}|} \quad (2.23b)$$

$$m_{rec,ss} = k_{rec}V_{rec}\sqrt{|p_d - p_s|} \quad (2.23c)$$

$$\tau_{comp} = \sigma r \omega m_c, \quad (2.23d)$$

where k_{in} , k_{out} , and k_{rec} are the inlet, outlet, and recycle valve gains; p_{in} and p_{out} are the inlet and outlet pressures; r is the impeller diameter and σ is the slip factor.

2.3.2 Compressor Maps, Characteristics and Efficiency

The operation of a compressor can be identified by studying the compressor maps. A typical compressor map can be seen in Figure 2.2. Such maps provide information about the characteristics and performance of compressors. These maps are used to identify the operating point of a compressor and its efficiency at steady-state conditions. A single characteristic curve of a compressor describes the relationship between pressure ratio Π and the corrected mass flow rate m_c for a constant rotational speed ω . Figure 2.2 depicts curves with different operating speeds ω . The operation between surge and choke for a constant speed curve defines the operational range of this characteristic curve. The group of all the characteristics between minimum and maximum rotational speed constitutes the operational domain or feasible window of the operation of a compressor. The compressor cannot operate beyond the limits of this window due to physical, safety, power and mechanical constraints.

Surge is the phenomenon of reverse flow in the compressor, and it occurs when the machine compresses gas to high pressures at low flow rates. Not only does it reduce both performance and efficiency, but it can also cause damage to the compressor or other auxiliaries (Cortinovis et al., 2015). Additionally, it causes vibrations due to the reversal of the flow resulting in unacceptable noise levels. Due to undesired consequences of the surge effect, compressors have to operate at a reduced operational window. This operational window is restricted

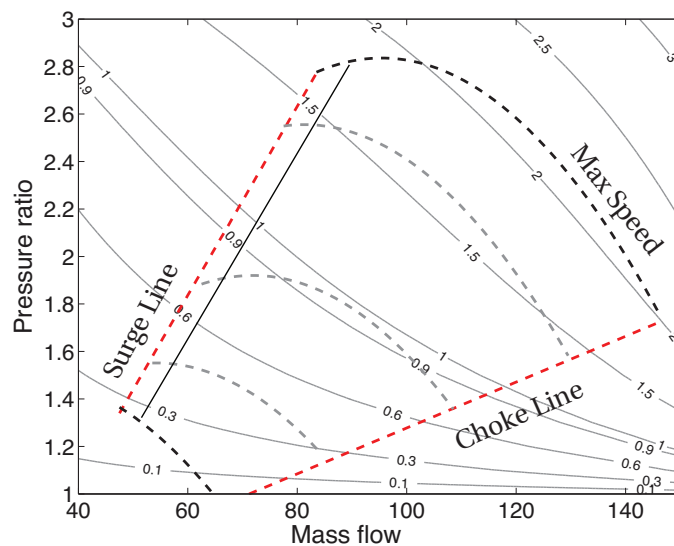


Figure 2.2 – Compressor operating constraints and cost contour curves (gray line). Upper and lower bound on speed (dashed black), operating points for three constant speeds (dashed gray), choke and surge line (dashed red) and surge control line parallel to surge line (black).

by a left boundary, known as control surge line. This line is located to the right of the surge line and both lines are separated by a safety margin.

Choke or stonewall region is located at the right region of a performance map. A choke point is described as the operational point of a compressor which is reached at the maximum flow rate and the minimum head for a fixed speed curve. It can also cause serious damage to the rotors and blades of multi-stage centrifugal or axial compressors (Bloch, 2006). Dixon and Hall (2013) presented a detailed explanation of the physical constraints of surge and choke with the use of fundamental aerodynamic and thermodynamic equations. The upper and lower bounds of the speed of a compressor map are defined by the specifications of the prime mover of the compressor, i.e. electric motor or gas turbine.

According to the manufacturer and the application, there are other descriptions of compressors maps which may consider for example discharge pressure, or isentropic or polytropic head on the vertical axis (Dixon and Hall, 2013). The pressure ratio Π is modeled as a polynomial function of ω and m_c . Likewise, the polytropic efficiency η_p is modeled as a polynomial function of ω and Π Cortinovis et al. (2016):

$$\Pi = \alpha_1 + \alpha_2\omega + \alpha_3 m_c + \alpha_4\omega m_c + \alpha_5\omega^2 + \alpha_6 m_c^2 \quad (2.24a)$$

$$\eta_p = \beta_1 + \beta_2\omega + \beta_3\Pi + \beta_4\omega\Pi + \beta_5\omega^2 + \beta_6\Pi^2. \quad (2.24b)$$

These compressor maps are typically provided by the manufacturer or they can be identified based on historical data Cortinovis et al. (2016).

The consideration of both the operating point, which describes pressure and flow rate, and the efficiency of the compressor can be used to estimate the power consumed by the gas at specific conditions. The shaft power is calculated as (Abbaspour et al., 2005)

$$\Phi = \frac{y_p}{\eta_p} m_c, \quad (2.25a)$$

where y_p is the polytropic head:

$$y_p = \frac{Z_{in}RT_{in}}{M_W} \frac{n_v}{n_v - 1} \left[\Pi^{\frac{n_v-1}{n_v}} - 1 \right]. \quad (2.25b)$$

Here, Z_{in} is the inlet compressibility factor, T_{in} the suction temperature, M_W the molecular weight of the gas mixture assumed to be constant, and n_v the polytropic exponent.

3 Modifier Adaptation for Interconnected Systems using Global Modifiers

This chapter is based on:

R. Schneider, P. Milosavljevic, and D. Bonvin. Distributed modifier-adaptation schemes for the real-time optimization of uncertain interconnected systems. *International Journal of Control*, pages 1–14, 2017.

This chapter deals with two distributed RTO schemes for systems that are composed of *interconnected subsystems*. This term emphasizes the dynamic interactions between the subsystems, which are typically caused by an exchange of material, energy or information. Both schemes are based on the MA framework. In contrast to ISOPE, MA consistently handles the uncertain cost function and the uncertain inequality constraints without requiring parameter estimation (Marchetti et al., 2009, 2016). Nevertheless, it is able to drive the plant to optimality (Marchetti et al., 2009). Here, these features are leveraged for interconnected systems.

In particular, both distributed MA algorithms use plant measurements in place of an interconnection model. This way, no coordinator is needed, and the subsystem optimizers do not have to share their local models. This feature makes these algorithms particularly attractive for privacy-sensitive applications, such as site-wide optimization in industrial parks operated by a consortium of competing companies (Wenzel et al., 2016). However, due to the absence or uncertainty in the interconnection model, both algorithms rely on the use of *global modifiers*. This means that the plant gradients of one subsystem with respect to the inputs of the other subsystems are required. Finally, it is shown for both schemes that, upon convergence, the computed inputs optimize the steady-state performance of the interconnected plant.

In many cases, *centralized* control of systems with interconnected subsystems may be difficult or undesirable. This is because (i) the individual subsystems may be physically at different locations as in hydroelectric power networks, (ii) of a prohibitively high computational complexity as in large chemical plants, or (iii) of concerns about the dependency of the whole

system on a single controller. These issues are well known, and various alternative control architectures have been developed (see Scattolini (2009) for a recent survey). These alternative control architectures include *decentralized* approaches, in which every subsystem is controlled by a dedicated controller that computes the local inputs based only on measurements from that subsystem. Another alternative is *distributed* control schemes. Similarly to the decentralized case, each subsystem is controlled by a dedicated controller, but the subsystem controllers might exchange information among each others. If, in addition, the local control actions are coordinated by a distinct coordinating unit, the resulting architecture is called *hierarchical control* (Scattolini, 2009) or *coordinated control* (Al-Gherwi et al., 2011). Note that, since the presence of a coordinator is often considered by default, many researchers call their coordinated control schemes simply distributed as well. These different control architectures can be found at all levels of the control pyramid, from the base and advanced control layers to the steady-state optimization and scheduling levels at the top. In the context of interconnected systems, distributed and hierarchical architectures have been found to be particularly advantageous when combined with methods relying on mathematical model-based optimization. These methods are typically applied at the upper levels of the control pyramid. For example, at the level of model predictive control, a variety of distributed and hierarchical methods exist for control (Christofides et al., 2013), state estimation (Schneider et al., 2015), and output feedback (Schneider et al., 2014). These techniques typically draw upon dynamic process models to solve dynamic optimization problems. At the top level, methods to decompose and distribute conventional plant-wide optimization problems have been suggested, e.g., for the tasks of scheduling (Xu et al., 2012) and steady-state optimization (Bryds et al., 1989). At this level, steady-state models give rise to static optimization problems.

The chapter is organized as follows. Section 3.1 presents the plant optimization problem for systems with explicit internal variables. In Section 3.2, a detailed model for interconnected systems is introduced and utilized to obtain a KKT point of the plant optimization problem via a centralized MA scheme. In Section 3.3, the two distributed modifier-adaptation algorithms are proposed and analyzed. Their main features are illustrated in Section 3.4 on a simple numerical example. Finally, Section 3.5 concludes the chapter with a summary and outlook on improving directions.

3.1 Problem Formulation

In contrast to the plant problem formulation given in Section 2.1, this section introduces the general steady-state optimization problem for a plant with internal variables. The cost function and constraints depend not only on the inputs but also on the outputs and the internal variables. However, provided that these output and internal variables are unique for every input vector, knowledge of their optimal values is not required for optimal plant operation. This section shows that it suffices to deal with the inputs to the plant, thus transforming the problem into the standard plant optimization problem (2.1), known from the RTO literature.

3.1.1 Plant Optimization Problem Including Output and Internal Variables

Consider the plant with the outputs \mathbf{y}_p and the internal variables \mathbf{v}_p as shown in Figure 3.1. In order to operate this plant optimally, one wishes to determine the optimal steady-state plant variables, that is, the minimizers of the following plant optimization problem

$$(\mathbf{u}_p^*, \mathbf{v}_p^*, \mathbf{y}_p^*) = \underset{\mathbf{u}, \mathbf{v}_p, \mathbf{y}_p}{\operatorname{argmin}} \Phi_p(\mathbf{u}, \mathbf{v}_p, \mathbf{y}_p) \quad (3.1a)$$

$$\text{s.t. } \mathbf{y}_p = \mathbf{F}_p(\mathbf{u}, \mathbf{v}_p), \quad (3.1b)$$

$$\mathbf{G}_p(\mathbf{u}, \mathbf{v}_p, \mathbf{y}_p) \leq \mathbf{0}, \quad (3.1c)$$

$$\mathbf{v}_p = \mathbf{H}_p(\mathbf{y}_p), \quad (3.1d)$$

where the subscript p denotes quantities related to the plant, and where $\mathbf{u} \in \mathbb{R}^{n_u}$, $\mathbf{v}_p \in \mathbb{R}^{n_v}$ and $\mathbf{y}_p \in \mathbb{R}^{n_y}$ are the steady-state plant inputs, internal variables and outputs, respectively. At steady state, the outputs are defined by the equality constraint (3.1b), and their relationship with the internal variables is represented by the n_v equality constraints (3.1d). Moreover, the n_G inequality constraints (3.1c) reflect the limits within which the plant can be operated safely.

From a practical perspective, the optimal values of the internal variables \mathbf{v}_p and \mathbf{y}_p are less important than the optimal values of the inputs \mathbf{u} . This is because only these inputs can be adjusted to optimize the steady-state performance of the plant. For this reason, the plant optimization problem (3.1) is often formulated differently, as shown in the next section.

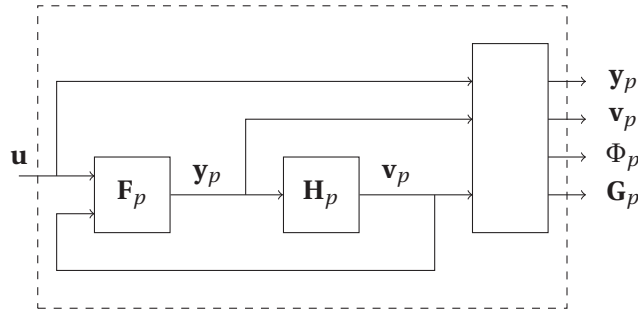


Figure 3.1 – Plant with the inputs \mathbf{u} , the outputs \mathbf{y}_p and the internal variables \mathbf{v}_p .

3.1.2 Standard Plant Optimization Problem

Since only the inputs \mathbf{u} of the plant can be adjusted, it is reasonable to simplify Problem (3.1) by eliminating the dependence of the cost and constraint functions on \mathbf{v}_p and \mathbf{y}_p . Mathematically, this is possible under the following assumption:

Assumption 3.1 (Unique plant steady state). *There exist unique and continuously differentiable functions $\mathbf{f}_p(\cdot)$ and $\mathbf{h}_p(\cdot)$ such that, for every admissible input vector \mathbf{u} , the equations*

Chapter 3. Modifier Adaptation for Interconnected Systems using Global Modifiers

(3.1b) and (3.1d) are equivalent to the equations

$$\mathbf{y}_p = \mathbf{f}_p(\mathbf{u}), \quad (3.2a)$$

$$\mathbf{v}_p = \mathbf{h}_p(\mathbf{u}). \quad (3.2b)$$

□

This assumption allows transforming the plant optimization problem with the output and internal variables (3.1) into the standard formulation, presented in Problem (2.1), frequently found in the RTO literature as described in the following Lemma:

Lemma 3.1 (Relationship to the standard plant optimization problem). *Let*

$$\phi_p(\mathbf{u}) := \Phi_p(\mathbf{u}, \mathbf{h}_p(\mathbf{u}), \mathbf{f}_p(\mathbf{u})), \quad (3.3a)$$

$$\mathbf{g}_p(\mathbf{u}) := \mathbf{G}_p(\mathbf{u}, \mathbf{h}_p(\mathbf{u}), \mathbf{f}_p(\mathbf{u})). \quad (3.3b)$$

Under Assumption 3.1, the solution to Problem (3.1) consists of the solution to the standard plant optimization problem (2.1)

$$\mathbf{u}_p^* = \arg \min_{\mathbf{u}} \phi_p(\mathbf{u}) \quad (3.4a)$$

$$s.t. \mathbf{g}_p(\mathbf{u}) \leq \mathbf{0}, \quad (3.4b)$$

and the relations

$$\mathbf{y}_p^* = \mathbf{f}_p(\mathbf{u}_p^*), \quad (3.4c)$$

$$\mathbf{v}_p^* = \mathbf{h}_p(\mathbf{u}_p^*). \quad (3.4d)$$

Proof. The proof can be found in Appendix C.1. □

Lemma 3.1 shows that, in order to force the plant to operate at the optimal tuple $(\mathbf{u}_p^*, \mathbf{v}_p^*, \mathbf{y}_p^*)$, it suffices to compute and implement the optimal inputs \mathbf{u}_p^* . The optimal values of \mathbf{y}_p^* and \mathbf{v}_p^* will then be automatically enforced by the plant upon reaching steady state, cf. equations (3.4c) and (3.4d). Unfortunately, computing the optimal inputs is not straightforward, as will be seen in the next section.

3.2 Centralized MA for Interconnected Systems

As already mentioned, some or all of the plant functions $\Phi_p(\cdot, \cdot, \cdot)$, $\mathbf{F}_p(\cdot, \cdot)$, $\mathbf{G}_p(\cdot, \cdot, \cdot)$, and $\mathbf{H}_p(\cdot)$ in (3.1) are typically unknown in practice. Consequently, the functions $\mathbf{f}_p(\cdot)$ and $\mathbf{h}_p(\cdot)$, as well as the cost and constraint functions $\phi_p(\cdot)$ and $\mathbf{g}_p(\cdot)$ in (3.4) are also unknown. Hence, neither Problem (3.1) nor (3.4) can be solved directly. One approach to deal with this problem is the method of modifier adaptation. Here, MA performance is analyzed when utilizing a

model whose structure closely resembles the one of the plant shown in Figure 3.1. As a result, distributed modifier-adaptation will be proposed in the next section.

3.2.1 MA with a Model for the Output and Internal Variables

The standard MA approach presented in Section 2.2 is quite general. However, for plants with the structure shown in Figure 3.1 expliciting the inputs, outputs and internal variables, one often has models for the functions $\Phi_p(\cdot, \cdot, \cdot)$, $\mathbf{F}_p(\cdot, \cdot)$, $\mathbf{G}_p(\cdot, \cdot, \cdot)$, and $\mathbf{H}_p(\cdot)$, rather than for $\phi_p(\cdot)$ and $\mathbf{g}_p(\cdot)$. Figure 3.2 shows the corresponding model structure, which consists of the cost model $\Phi(\mathbf{u}, \mathbf{v}, \mathbf{y})$, the constraint models $\mathbf{G}(\mathbf{u}, \mathbf{v}, \mathbf{y}) \leq \mathbf{0}$, and the steady-state model

$$\mathbf{y} = \mathbf{F}(\mathbf{u}, \mathbf{v}), \quad (3.5a)$$

$$\mathbf{v} = \mathbf{H}(\mathbf{y}), \quad (3.5b)$$

where $\mathbf{y} \in \mathbb{R}^{n_y}$ and $\mathbf{v} \in \mathbb{R}^{n_v}$ are the modeled output and internal variables, respectively.

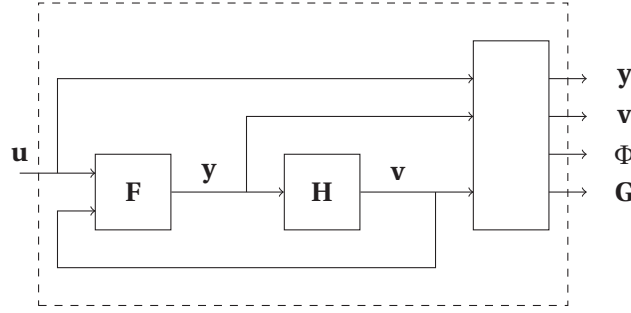


Figure 3.2 – Model of the plant with the inputs \mathbf{u} , the outputs \mathbf{y} and the internal variables \mathbf{v} .

When these models are available, one may want to formulate the modifier-adaptation algorithm directly in terms of these models. To do this, the following assumption is required:

Assumption 3.2 (Unique model steady state). *There exist unique and continuously differentiable functions $\mathbf{f}(\cdot)$ and $\mathbf{h}(\cdot)$ such that, for every admissible input vector \mathbf{u} , the equations (3.5a) and (3.5b) are equivalent to the equations*

$$\mathbf{y} = \mathbf{f}(\mathbf{u}), \quad (3.6a)$$

$$\mathbf{v} = \mathbf{h}(\mathbf{u}). \quad (3.6b)$$

□

Then, the MA algorithm is formulated, which uses a model for the output and internal variables, as follows:

$$\left(\mathbf{u}_{k+1}^*, \mathbf{v}_{k+1}^*, \mathbf{y}_{k+1}^* \right) = \underset{\mathbf{u}, \mathbf{v}, \mathbf{y}}{\operatorname{argmin}} \Phi(\mathbf{u}, \mathbf{v}, \mathbf{y}) + (\boldsymbol{\lambda}_k^\phi)^\top (\mathbf{u} - \mathbf{u}_k) \quad (3.7a)$$

s.t.

$$\mathbf{y} = \mathbf{F}(\mathbf{u}, \mathbf{v}), \quad (3.7b)$$

$$\mathbf{G}(\mathbf{u}, \mathbf{v}, \mathbf{y}) + \boldsymbol{\epsilon}_k^g + (\boldsymbol{\lambda}_k^g)^\top (\mathbf{u} - \mathbf{u}_k) \leq \mathbf{0}, \quad (3.7c)$$

$$\mathbf{v} = \mathbf{H}(\mathbf{y}), \quad (3.7d)$$

where the modifiers are computed according to (2.5a)-(2.5c) with the model functions $\phi(\cdot)$ and $\mathbf{g}(\cdot)$ defined, in analogy to (3.3a) and (3.3b), as

$$\phi(\mathbf{u}) := \Phi(\mathbf{u}, \mathbf{h}(\mathbf{u}), \mathbf{f}(\mathbf{u})), \quad (3.7e)$$

$$\mathbf{g}(\mathbf{u}) := \mathbf{G}(\mathbf{u}, \mathbf{h}(\mathbf{u}), \mathbf{f}(\mathbf{u})). \quad (3.7f)$$

It may be surprising that (3.7a) contains modifier terms for \mathbf{u} but not for the other optimization variables \mathbf{v} or \mathbf{y} . However, this problem formulation enables us to establish the equivalence between this algorithm and the standard MA scheme (2.4). Before doing so, note that similar to standard MA, the existence of a solution at every iteration k is assumed. Also, the filter as in (2.6) is applied as

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \mathbf{K}(\mathbf{u}_{k+1}^* - \mathbf{u}_k) \quad (3.7g)$$

Now, the equivalence of this algorithm with the standard MA algorithm is established. We state the following result as a special case, namely for $N = 1$, of the proof of Proposition 3.1 that will be discussed latter.

Corollary 3.1 (Equivalence with standard MA). *Under Assumption 3.2, the optimal inputs \mathbf{u}^* obtained from (3.7) are equivalent to those obtained from (2.4).*

Even though they are equivalent, the MA formulations (3.7) and (2.4) serve different purposes in this work: As shown below, the formulation (3.7) will enable us to use intuitive models of interconnected systems. This paves the way for distributed MA schemes, whose ability to converge to a KKT point of the plant despite plant-model mismatch, will be proven in Section 3.3 with the help of Problem 2.4.

3.2.2 Modeling an Interconnected System

For interconnected systems, the model with output and internal variables can be refined further. In this section, the model of the plant is discussed under the assumption that it consists of N interconnected subsystems $i \in \mathcal{N} = \{1, \dots, N\}$ as shown in Figure 3.3. Each of these subsystem models, shown in more detail in Figure 3.4, has the n_{u_i} inputs \mathbf{u}_i , the n_{v_i} interconnection inputs \mathbf{v}_i , the n_{y_i} outputs \mathbf{y}_i . In addition, there are n_{y_i} output equations and n_{G_i}

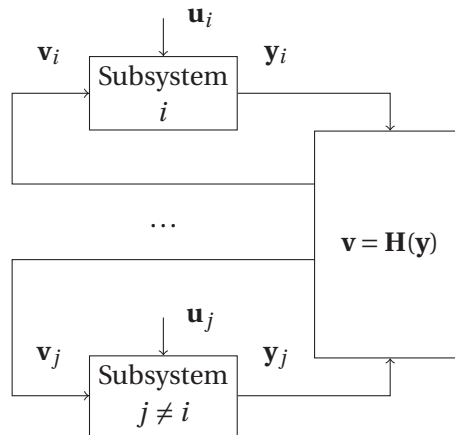


Figure 3.3 – Model of a plant composed of interconnected subsystems. The interconnection variables are the inputs $\mathbf{v} = \text{col}(\mathbf{v}_1, \dots, \mathbf{v}_N)$.

inequality constraints modeled as

$$\mathbf{y}_i = \mathbf{F}_i(\mathbf{u}_i, \mathbf{v}_i), \quad (3.8a)$$

$$\mathbf{G}_i(\mathbf{u}_i, \mathbf{v}_i, \mathbf{y}_i) \leq \mathbf{0}, \quad \forall i \in \mathcal{N}. \quad (3.8b)$$

Collectively, the vectors and functions of all subsystem models correspond to the vectors and functions known from (3.5a) in the following way:

$$\mathbf{u} = \text{col}(\mathbf{u}_1, \dots, \mathbf{u}_N), \quad (3.8c)$$

$$\mathbf{v} = \text{col}(\mathbf{v}_1, \dots, \mathbf{v}_N), \quad (3.8d)$$

$$\mathbf{y} = \text{col}(\mathbf{y}_1, \dots, \mathbf{y}_N), \quad (3.8e)$$

$$\mathbf{F}(\mathbf{u}, \mathbf{v}) = \text{col}(\mathbf{F}_1(\mathbf{u}_1, \mathbf{v}_1), \dots, \mathbf{F}_N(\mathbf{u}_N, \mathbf{v}_N)), \quad (3.8f)$$

$$\mathbf{G}(\mathbf{u}, \mathbf{v}, \mathbf{y}) = \text{col}(\mathbf{G}_1(\mathbf{u}_1, \mathbf{v}_1, \mathbf{y}_1), \dots, \mathbf{G}_N(\mathbf{u}_N, \mathbf{v}_N, \mathbf{y}_N)). \quad (3.8g)$$

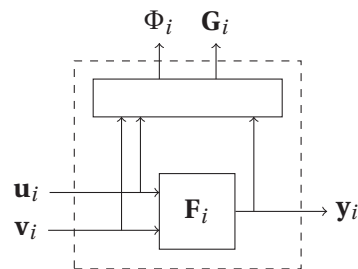


Figure 3.4 – Model of the i -th subsystem.

Chapter 3. Modifier Adaptation for Interconnected Systems using Global Modifiers

Notice that the variables \mathbf{v} , introduced as *internal variables* in the previous section, here serve the purpose of connecting the different subsystems

$$\mathbf{v} = \mathbf{H}(\mathbf{y}). \quad (3.8h)$$

Hence, they will be denoted as *interconnection variables*. From Figure 3.3, it can be seen that the interconnection variables \mathbf{v}_i of the subsystem model i generally depend on the outputs of at least some other subsystem model $j \neq i$. Thus, unless the subsystems are decoupled, (3.8h) *cannot* be written as

$$\mathbf{v}_i = \mathbf{H}_i(\mathbf{y}_i), \quad \forall i \in \mathcal{N}. \quad (3.9)$$

Furthermore, the model $\Phi(\cdot, \cdot, \cdot)$ of the plant cost $\Phi_p(\cdot, \cdot, \cdot)$ is assumed to be separable,

$$\Phi(\mathbf{u}, \mathbf{v}, \mathbf{y}) = \sum_{i \in \mathcal{N}} \Phi_i(\mathbf{u}_i, \mathbf{v}_i, \mathbf{y}_i). \quad (3.10)$$

3.2.3 Proposed Centralized MA Scheme

Here, a centralized MA scheme for interconnected systems is formulated using the model derived in (3.7). The following optimization problem is solved at the k -th iteration

$$\left(\mathbf{u}_{k+1}^*, \mathbf{v}_{k+1}^*, \mathbf{y}_{k+1}^* \right) = \underset{\mathbf{u}, \mathbf{v}, \mathbf{y}}{\operatorname{argmin}} \sum_{i \in \mathcal{N}} \left(\Phi_i(\mathbf{u}_i, \mathbf{v}_i, \mathbf{y}_i) + (\boldsymbol{\lambda}_{i,k}^\phi)^\top (\mathbf{u}_i - \mathbf{u}_{i,k}) \right) \quad (3.11a)$$

s.t. $\forall i \in \mathcal{N}$:

$$\mathbf{y}_i = \mathbf{F}_i(\mathbf{u}_i, \mathbf{v}_i), \quad (3.11b)$$

$$\mathbf{G}_i(\mathbf{u}_i, \mathbf{v}_i, \mathbf{y}_i) + \boldsymbol{\epsilon}_{i,k}^g + (\boldsymbol{\lambda}_{i,k}^g)^\top (\mathbf{u} - \mathbf{u}_k) \leq \mathbf{0}, \quad (3.11c)$$

$$\mathbf{v} = \mathbf{H}(\mathbf{y}), \quad (3.11d)$$

where $(\boldsymbol{\lambda}_{i,k}^\phi)^\top$ is the n_{u_i} -dimensional i -th block-column of the modifiers $(\boldsymbol{\lambda}_k^\phi)^\top$, $\boldsymbol{\epsilon}_{i,k}^g$ is the n_{G_i} -dimensional i -th block-row of the modifiers $\boldsymbol{\epsilon}_k^g$, and $(\boldsymbol{\lambda}_{i,k}^g)^\top$ is the $(n_{G_i} \times n_u)$ -dimensional i -th block-row of the modifiers $(\boldsymbol{\lambda}_k^g)^\top$. The modifiers are computed according to (2.5a)-(2.5c) with the model functions $\phi(\cdot)$ and $\mathbf{g}(\cdot)$ defined, in analogy to (3.7e) and (3.7f) and using the special model structure (3.8) and (3.10), as

$$\phi(\mathbf{u}) := \Phi(\mathbf{u}, \mathbf{h}(\mathbf{u}), \mathbf{f}(\mathbf{u})) \quad (3.11e)$$

$$= \sum_{i \in \mathcal{N}} \Phi_i(\mathbf{u}_i, \mathbf{h}_i(\mathbf{u}), \mathbf{f}_i(\mathbf{u})), \quad (3.11f)$$

$$\mathbf{g}(\mathbf{u}) := \mathbf{G}(\mathbf{u}, \mathbf{h}(\mathbf{u}), \mathbf{f}(\mathbf{u})) \quad (3.11g)$$

$$= \operatorname{col}(\mathbf{G}_1(\mathbf{u}_1, \mathbf{h}_1(\mathbf{u}), \mathbf{f}_1(\mathbf{u})), \dots, \mathbf{G}_N(\mathbf{u}_N, \mathbf{h}_N(\mathbf{u}), \mathbf{f}_N(\mathbf{u}))), \quad (3.11h)$$

3.3. Distributed MA without Interconnection Model

where $\mathbf{f}_i(\cdot)$ and $\mathbf{h}_i(\cdot)$ are the components of the functions $\mathbf{f}(\cdot)$ and $\mathbf{h}(\cdot)$ defined in Assumption 3.2. Moreover, the existence of a solution at every iteration k is assumed. Also, the filter (3.7g) is applied. It can be shown that the fixed point is a KKT point of the plant optimization problem (3.4a) subject to (3.4b). In its simplest form, the algorithm proceeds as follows

Algorithm 3.1 : Centralised MA with Global Modifiers

Initialization: choose the diagonal filter matrix \mathbf{K} in (3.7g) with eigenvalues in the interval $(0, 1]$ and choose a feasible input vector \mathbf{u}_0 .

for $k = 0 \rightarrow \infty$ **do**:

1. Apply the inputs \mathbf{u}_k to the plant and wait for steady state.
2. Update the modifiers according to (2.5a)-(2.5c).
3. Solve the modified model-based optimization problem (3.11)
4. Filter the inputs according to (3.7g).

end

The property of this iterative procedure is provided by the following theorem.

Proposition 3.1 (KKT matching for Algorithm 3.1). *Let Assumption 3.2 hold and the sequence generated by Algorithm 3.1 converge, with $\mathbf{u}_\infty = \lim_{k \rightarrow \infty} \mathbf{u}_k$. Then, \mathbf{u}_∞ is a KKT point for the plant optimization problem (3.4a) subject to (3.4b).*

Proof. The proof can be found in Appendix C.2. □

Remark 3.1 (Mismatch in interconnection variables). *Even when the fixed point \mathbf{u}_∞ of the MA scheme is an optimal input vector for the plant, the converged outputs $\mathbf{y}_\infty := \lim_{k \rightarrow \infty} \mathbf{y}_k$ and interconnection variables $\mathbf{v}_\infty := \lim_{k \rightarrow \infty} \mathbf{v}_k$ are generally different from the plant values \mathbf{y}_p^* and \mathbf{v}_p^* . The reason lies in the mismatch between the plant functions $\mathbf{f}_p(\cdot)$ and $\mathbf{h}_p(\cdot)$ and their model counterparts $\mathbf{f}(\cdot)$ and $\mathbf{h}(\cdot)$. Nevertheless, provided that the computed input vector \mathbf{u}_∞ is optimal for the plant, its implementation will directly result in the plant settling at the optimal plant values \mathbf{y}_p^* and \mathbf{v}_p^* , cf. (3.4c) and (3.4d).*

The centralized MA for interconnected systems derived in this section will be the basis for the distributed MA schemes developed next.

3.3 Distributed MA without Interconnection Model

For interconnected systems, centralized RTO schemes may not be desirable or applicable. For example, centralized computers to solve optimization problems may not be available or local subsystem operators may not want to disclose their models to competitors. In these and similar situations, distributed RTO schemes may be beneficial.

In order to derive such schemes, some assumptions regarding the inequality constraints are made that can be considered. In each of the following subsections, a particular distributed MA scheme is presented and analyzed. These schemes differ in the amount of measured information used. The order in which these schemes are presented is chosen to reflect the decreasing model complexity and increasing use of measurements, without implying that one algorithm is generally better than another.

Specifically, each subsection is structured as follows. Firstly, the sequence of optimization problems is stated mathematically. Secondly, a convergence result is given for every distributed MA scheme. Thirdly, the assumptions needed to implement and solve the corresponding optimization problem in a distributed way are presented. Finally, each resulting algorithm is summarized and its technical requirements and practical advantages are discussed.

3.3.1 Assumptions regarding the Inequality Constraints

To see why additional assumptions on the inequality constraints are required for distributed MA schemes, consider the structure of Problem (3.11): At the k -th iteration, Problem (3.11) consists in the minimization of the separable cost (3.11a) subject to a set of separable equality constraints (3.11b), a set of coupled inequality constraints (3.11c), and a set of coupled equality constraints (3.11d). Clearly, the coupled constraints (3.11c) and (3.11d) are an obstacle toward a distributed MA scheme.

In this section, the inequality constraints are simplified. First, note that the local first-order constraint modifiers ϵ_i^g of every subsystem can be determined from the local model and plant variables if the following assumption holds:

Assumption 3.3 (Local plant inequalities). *For each subsystem, the plant inequality constraints depend only on adjustable or measurable variables associated with that subsystem. In other words, the plant inequality constraints are of the form*

$$\mathbf{G}_p(\mathbf{u}, \mathbf{v}, \mathbf{y}) = \text{col}(\mathbf{G}_{p,1}(\mathbf{u}_1, \mathbf{v}_{p,1}, \mathbf{y}_{p,1}), \dots, \mathbf{G}_{p,N}(\mathbf{u}_N, \mathbf{v}_{p,N}, \mathbf{y}_{p,N})). \quad (3.12)$$

□

Second, the local inequality constraints (3.11c) are generally coupled across subsystems, that is, the inequality constraint for subsystem i depends on \mathbf{u}_j . To make the subsequent analysis tractable, it is assumed that the plant and model inequalities are decoupled, which will be the case under the following two assumptions:

Assumption 3.4 (Decoupled inequalities). *The off-diagonal derivatives of the plant inequali-*

3.3. Distributed MA without Interconnection Model

ties vanish despite the presence of interconnection variables:

$$\frac{\partial \mathbf{g}_{p,i}(\mathbf{u})}{\partial \mathbf{u}_j} = \frac{d\mathbf{G}_{p,i}(\mathbf{u}_i, \mathbf{h}_{p,i}(\mathbf{u}), \mathbf{f}_{p,i}(\mathbf{u}))}{d\mathbf{u}_j} = \mathbf{0}, \forall i \in \mathcal{N} \setminus j. \quad (3.13)$$

□

Before stating the benefit of these assumptions, a brief comment on Assumption 3.4 is in order. Effectively, Assumption 3.4 covers distributed RTO problems with general uncertain constraint functions that depend only on the local subsystem inputs \mathbf{u}_i and those local subsystem outputs $\mathbf{y}_{p,i}$ that are independent of $\mathbf{v}_{p,i}$. In other words, suppose that the local subsystem outputs $\mathbf{y}_{p,i}$ can be partitioned into $\mathbf{y}_{p,i}^A$ and $\mathbf{y}_{p,i}^B$ as follows:

$$\mathbf{y}_{p,i}^A = \mathbf{F}_{p,i}^A(\mathbf{u}_i, \mathbf{v}_{p,i}), \quad (3.14a)$$

$$\mathbf{y}_{p,i}^B = \mathbf{F}_{p,i}^B(\mathbf{u}_i), \quad (3.14b)$$

that is, the local subsystem outputs $\mathbf{y}_{p,i}^B$ are independent of $\mathbf{v}_{p,i}$ (and hence of $\mathbf{u}_{j \neq i}$). Then, Assumption 3.4 restricts the class of admissible plant inequality constraints to functions $\mathbf{G}_{p,i}$ that can be expressed as

$$\mathbf{G}_{p,i}(\mathbf{u}_i, \mathbf{y}_{p,i}^B) \leq \mathbf{0}. \quad (3.15)$$

Inequality constraints of this form can be found in important RTO problem classes. In process systems engineering, for example, they can be used to handle valve stiction by considering constraints on valve positions $\mathbf{y}_{p,i}$ which may be uncertain functions of the valve position set-points \mathbf{u}_i . Note that, uncertain inequality constraints are not considered in ISOPE algorithms for interconnected systems (Brdyś and Tatjewski, 2005).

As a consequence of Assumptions 3.3-3.4, the first-order constraint modifier blocks $\lambda_{ij,k}^g$ will be zero for every $j \neq i$, and the inequality constraints (3.11c) reduce to

$$\mathbf{G}_i(\mathbf{u}_i, \mathbf{v}_i, \mathbf{y}_i) + \boldsymbol{\epsilon}_{i,k}^g + (\boldsymbol{\lambda}_{ii,k}^g)^\top (\mathbf{u}_i - \mathbf{u}_{i,k}) \leq \mathbf{0}, \quad (3.16)$$

where $(\boldsymbol{\lambda}_{i,k}^g)^\top$ is the $(n_{G_i} \times n_{u_i})$ -dimensional i -th diagonal block of $(\boldsymbol{\lambda}_k^g)^\top$.

There are also viable alternatives to handle distributed RTO problems whose inequality constraints do not satisfy the above assumptions. Instead of imposing these constraints as hard constraints, a penalty or barrier function of their violation can be added to the original costs of the plant and the model, as proposed in different contexts by Srinivasan et al. (2008); Ellis et al. (1988).

While such soft constraints can be more general and possibly coupled, they may increase the number of necessary RTO iterations until convergence is achieved compared to the case of hard constraints.

3.3.2 Distributed MA without Interconnection Model

The algorithm in this section exploits the fact that MA is quite flexible with model uncertainty even if that is uncertainty in the interconnection model. In particular, if the plant measurements \mathbf{v}_p are available at every iteration, one do not need to use the interconnection model (3.11d) to link the model variables \mathbf{y} and \mathbf{v} . Instead, the model variables \mathbf{v} can be replaced by the plant measurements \mathbf{v}_p at every iteration. This results in the following optimization problems at the k -th iteration:

$$(\mathbf{u}_{k+1}^*, \mathbf{y}_{k+1}^*) = \underset{\mathbf{u}, \mathbf{y}}{\operatorname{argmin}} \sum_{i \in \mathcal{N}} \left(\Phi_i(\mathbf{u}_i, \mathbf{v}_{p,i,k}, \mathbf{y}_i) + (\boldsymbol{\lambda}_{i,k}^\phi)^\top (\mathbf{u}_i - \mathbf{u}_{i,k}) \right) \quad (3.17a)$$

s.t. $\forall i \in \mathcal{N}$:

$$\mathbf{y}_i = \mathbf{F}_i(\mathbf{u}_i, \mathbf{v}_{p,i,k}), \quad (3.17b)$$

$$\mathbf{G}_i(\mathbf{u}_i, \mathbf{v}_{p,i,k}, \mathbf{y}_i) + \boldsymbol{\epsilon}_{i,k}^g + (\boldsymbol{\lambda}_{i,k}^g)^\top (\mathbf{u} - \mathbf{u}_k) \leq \mathbf{0}, \quad (3.17c)$$

where again $(\boldsymbol{\lambda}_{i,k}^\phi)^\top$ is the n_{u_i} -dimensional i -th block-column of $(\boldsymbol{\lambda}_k^\phi)^\top$, $\boldsymbol{\epsilon}_{i,k}^g$ is the n_{G_i} -dimensional i -th block-row of $\boldsymbol{\epsilon}_k^g$, and $(\boldsymbol{\lambda}_{i,k}^g)^\top$ is the $(n_{G_i} \times n_u)$ -dimensional i -th block-row of $(\boldsymbol{\lambda}_k^g)^\top$. Here, the special model structure (3.8) and (3.10) and the plant measurements \mathbf{v}_p are used to define the model functions $\phi(\cdot)$ and $\mathbf{g}(\cdot)$ as

$$\phi(\mathbf{u}) := \Phi(\mathbf{u}, \mathbf{v}_{p,k}, \mathbf{F}(\mathbf{u}, \mathbf{v}_{p,k})) \quad (3.17d)$$

$$= \sum_{i \in \mathcal{N}} \Phi_i(\mathbf{u}_i, \mathbf{v}_{p,i,k}, \mathbf{F}_i(\mathbf{u}_i, \mathbf{v}_{p,i,k})), \quad (3.17e)$$

$$\mathbf{g}(\mathbf{u}) := \mathbf{G}(\mathbf{u}, \mathbf{v}_{p,k}, \mathbf{F}(\mathbf{u}, \mathbf{v}_{p,k})) \quad (3.17f)$$

$$= \operatorname{col} \left(\mathbf{G}_1(\mathbf{u}_1, \mathbf{v}_{p,1,k}, \mathbf{F}_1(\mathbf{u}_1, \mathbf{v}_{p,1,k})), \dots, \mathbf{G}_N(\mathbf{u}_N, \mathbf{v}_{p,N,k}, \mathbf{F}_N(\mathbf{u}_N, \mathbf{v}_{p,N,k})) \right). \quad (3.17g)$$

Moreover, the existence of a solution of (3.17) at every iteration k is required. Also, the input filter is applied. Note that, MA scheme (3.17) does not require Assumption 3.2 to show that any fixed point is a KKT point of the plant optimization problem (3.4a) subject to (3.4b).

Problem (3.17) is tackled in a distributed way. Due to the absence of an interconnection model, there are no equalities to be satisfied. Hence, one only need to impose eq. (3.13) in Assumptions 3.4 and Assumption 3.3, or replace the hard inequality constraints by soft constraints, to compute the solution to Problem (3.17) by solving in parallel the following subproblems (for $i \in \mathcal{N}$) at the k -th iteration:

$$\left(\mathbf{u}_{i,k+1}^*, \mathbf{y}_{i,k+1}^* \right) = \underset{\mathbf{u}_i, \mathbf{y}_i}{\operatorname{argmin}} \left(\Phi_i(\mathbf{u}_i, \mathbf{v}_{p,i,k}, \mathbf{y}_i) + (\boldsymbol{\lambda}_{i,k}^\phi)^\top (\mathbf{u}_i - \mathbf{u}_{i,k}) \right) \quad (3.18a)$$

3.3. Distributed MA without Interconnection Model

s.t.

$$\mathbf{y}_i = \mathbf{F}_i(\mathbf{u}_i, \mathbf{v}_{p,i,k}), \quad (3.18b)$$

$$\mathbf{G}_i(\mathbf{u}_i, \mathbf{v}_{p,i,k}, \mathbf{y}_i) + \boldsymbol{\epsilon}_{i,k}^g + (\boldsymbol{\lambda}_{ii,k}^g)^\top (\mathbf{u}_i - \mathbf{u}_{i,k}) \leq \mathbf{0}. \quad (3.18c)$$

If one also parallelize the filtering step (3.7g) by enforcing the filter gain matrix to be block-diagonal, that is, $\mathbf{K} = \text{blkdiag}(\mathbf{K}_1, \dots, \mathbf{K}_N)$, the following distributed modifier-adaptation algorithm that does not use any interconnection model is obtained:

Algorithm 3.2 : Distributed MA without interconnection model

Initialization: Choose a block-diagonal and nonsingular filter gain matrix \mathbf{K} and a feasible input vector $\mathbf{u}(0)$.

for $k = 0 \rightarrow \infty$ and, in parallel for all $i \in \mathcal{N}$, do:

1. Apply the inputs $\mathbf{u}_{i,k}$ to the plant and wait for steady state.
2. Measure the plant constraint values $\mathbf{g}_{p,i}(\mathbf{u}_k)$ and the interconnection variables $\mathbf{v}_{p,i,k}$. Estimate the gradients $\left. \frac{d\phi_p(\mathbf{u})}{d\mathbf{u}_i} \right|_{\mathbf{u}=\mathbf{u}_k}$ and $\left. \frac{d\mathbf{g}_p(\mathbf{u})}{d\mathbf{u}_i} \right|_{\mathbf{u}=\mathbf{u}_k}$.
3. Update the modifiers according to (2.5a)-(2.5c).
4. Compute the solution $\mathbf{u}_{i,k+1}^*$ to the optimization problems (3.18).
5. Filter the input vector $\mathbf{u}_{i,k+1}^*$ as

$$\mathbf{u}_{i,k+1} = \mathbf{u}_{i,k} + \mathbf{K}_i (\mathbf{u}_{i,k+1}^* - \mathbf{u}_{i,k}). \quad (3.19)$$

end

Hence, by using measurements of the interconnection variables \mathbf{v}_p instead of the interconnection model (3.11d), fully parallelizable algorithm is obtained as illustrated in Figure 3.5. Note that there is no information exchange between the subsystem optimizers. This is because only the local functions $\Phi_i(\cdot, \cdot, \cdot)$, $\mathbf{F}_i(\cdot, \cdot)$, and $\mathbf{G}_i(\cdot, \cdot, \cdot)$ are required to compute the model gradients. On the other hand, if a good interconnection model is available, disregarding the interconnection information may increase the number of iterations until convergence is achieved.

The property of Algorithm 3.2 is provided by the following theorem.

Proposition 3.2 (KKT matching for Algorithm 3.2). *Let the sequence generated by Algorithm 3.2 converge, with $\mathbf{u}_\infty = \lim_{k \rightarrow \infty} \mathbf{u}_k$. Then, \mathbf{u}_∞ is a KKT point for the plant optimization problem (3.4a) subject to (3.4b).*

Proof. The proof can be found in Appendix C.3. □

However, note that the current algorithm is not completely decentralized, since each sub-

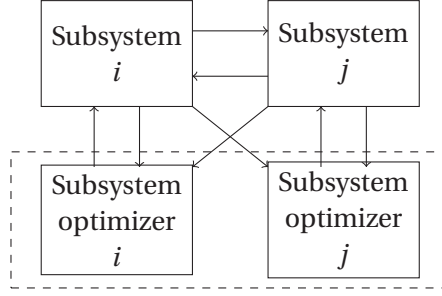


Figure 3.5 – Coordination-free structure of Algorithms 3.2 and 3.3.

system optimizer still relies on information from other subsystems in order to determine the plant gradients with respect to the local inputs. Namely, mismatch in interconnection model is compensated by plant gradient estimation of one system with respect to the inputs of other systems. For large scale systems as well as systems with low settling time and different dynamics this can be very expensive, resulting in slow convergence and large number of setpoint changes.

3.3.3 Distributed MA without Interconnection and Output Models

Let us go one step further and assume that there is neither an interconnection model nor a model for the outputs \mathbf{y} . Then, the plant measurements \mathbf{v}_p and \mathbf{y}_p are utilized instead and the following optimization problem is proposed at the k -th iteration:

$$\mathbf{u}_{k+1}^* = \underset{\mathbf{u}}{\operatorname{argmin}} \sum_{i \in \mathcal{N}} \left(\Phi_i(\mathbf{u}_i, \mathbf{v}_{p,i,k}, \mathbf{y}_{p,i,k}) + (\boldsymbol{\lambda}_{i,k}^\phi)^\top (\mathbf{u}_i - \mathbf{u}_{i,k}) \right) \quad (3.20a)$$

$$\text{s.t. } \mathbf{G}_i(\mathbf{u}_i, \mathbf{v}_{p,i,k}, \mathbf{y}_{p,i,k}) + \boldsymbol{\epsilon}_{i,k}^g + (\boldsymbol{\lambda}_{i,k}^g)^\top (\mathbf{u} - \mathbf{u}_k) \leq \mathbf{0}, \quad \forall i \in \mathcal{N}, \quad (3.20b)$$

where again $(\boldsymbol{\lambda}_{i,k}^\phi)^\top$ is the n_{u_i} -dimensional i -th block-column of $(\boldsymbol{\lambda}_k^\phi)^\top$, $\boldsymbol{\epsilon}_{i,k}^g$ is the n_{G_i} -dimensional i -th block-row of $\boldsymbol{\epsilon}_k^g$, and $(\boldsymbol{\lambda}_{i,k}^g)^\top$ is the $(n_{G_i} \times n_u)$ -dimensional i -th block-row of $(\boldsymbol{\lambda}_k^g)^\top$. Here, the special model structure (3.8) and (3.10) and the plant measurements \mathbf{v}_p and \mathbf{y}_p are used to define the model functions $\phi(\cdot)$ and $\mathbf{g}(\cdot)$ as

$$\phi(\mathbf{u}) := \Phi(\mathbf{u}, \mathbf{v}_{p,k}, \mathbf{y}_{p,k}) \quad (3.20c)$$

$$= \sum_{i \in \mathcal{N}} \Phi_i(\mathbf{u}_i, \mathbf{v}_{p,i,k}, \mathbf{y}_{p,i,k}), \quad (3.20d)$$

$$\mathbf{g}(\mathbf{u}) := \mathbf{G}(\mathbf{u}, \mathbf{v}_{p,k}, \mathbf{y}_{p,k}) \quad (3.20e)$$

$$= \operatorname{col}(\mathbf{G}_1(\mathbf{u}_1, \mathbf{v}_{p,1,k}, \mathbf{y}_{p,1,k}), \dots, \mathbf{G}_N(\mathbf{u}_N, \mathbf{v}_{p,N,k}, \mathbf{y}_{p,N,k})). \quad (3.20f)$$

Moreover, the existence of a solution at every iteration k is required. Also, the filter (3.7g) is applied.

This approach does not require models of the interconnections and outputs. Instead,

3.3. Distributed MA without Interconnection Model

measurements of the outputs \mathbf{y}_p and of the interconnection variables \mathbf{v}_p are used. As with distributed MA Algorithm 3.2, the computation of the solution can be fully parallelized, provided that (i) the plant inequality constraints are decoupled, that is, under Assumptions 3.3 and eq. (3.13) in Assumptions 3.4, (ii) the plant gradients are available, and (iii) \mathbf{K} is a block-diagonal filter gain matrix. Recall that, as an alternative to imposing Assumptions 3.3 and 3.4, the inequality constraints can be treated as soft constraints via penalty or barrier terms in the cost. In any case, the solution to Problem (3.20) is obtained by solving in parallel the following subproblems (for $i \in \mathcal{N}$) at the k -th iteration:

$$\mathbf{u}_{i,k+1}^* = \underset{\mathbf{u}_i}{\operatorname{argmin}} \left(\Phi_i(\mathbf{u}_i, \mathbf{v}_{p,i,k}, \mathbf{y}_{p,i,k}) + (\boldsymbol{\lambda}_{i,k}^\phi)^\top (\mathbf{u}_i - \mathbf{u}_{i,k}) \right) \quad (3.21a)$$

$$\text{s.t. } \mathbf{G}_i(\mathbf{u}_i, \mathbf{v}_{p,i,k}, \mathbf{y}_{p,i,k}) + \boldsymbol{\epsilon}_{i,k}^g + (\boldsymbol{\lambda}_{i,k}^g)^\top (\mathbf{u}_i - \mathbf{u}_{i,k}) \leq \mathbf{0}. \quad (3.21b)$$

This results in the parallel Algorithm 3.3.

Algorithm 3.3 : Distributed MA without interconnection and output models

Initialization: Choose a block-diagonal and nonsingular filter gain matrix \mathbf{K} and a feasible input vector $\mathbf{u}(0)$.

for $k = 0 \rightarrow \infty$ and, in parallel for all $i \in \mathcal{N}$, do:

1. Apply the inputs $\mathbf{u}_{i,k}$ to the plant and wait for steady state.
2. Measure the plant constraint values $\mathbf{g}_{p,i}(\mathbf{u}_k)$, the interconnection variables $\mathbf{v}_{p,i,k}$ and the outputs $\mathbf{y}_{p,i,k}$. Estimate the gradients $\left. \frac{d\phi_p(\mathbf{u})}{d\mathbf{u}_i} \right|_{\mathbf{u}=\mathbf{u}_k}$ and $\left. \frac{d\mathbf{g}_p(\mathbf{u})}{d\mathbf{u}_i} \right|_{\mathbf{u}=\mathbf{u}_k}$.
3. Update the modifiers according to (2.5a)-(2.5c).
4. Obtain the solution $\mathbf{u}_{i,k+1}^*$ to the optimization problems (3.21).
5. Filter the inputs $\mathbf{u}_{i,k+1}^*$ as

$$\mathbf{u}_{i,k+1} = \mathbf{u}_{i,k} + \mathbf{K}_i (\mathbf{u}_{i,k+1}^* - \mathbf{u}_{i,k}). \quad (3.22)$$

end

Just as in Algorithm 3.2, model does not need to be exchanged between the subsystem optimizers as illustrated in Figure 3.5.

The property of Algorithm 3.3 is provided by the following theorem.

Proposition 3.3 (KKT matching for Algorithm 3.3). *Let the sequence generated by Algorithm 3.3 converge, with $\mathbf{u}_\infty = \lim_{k \rightarrow \infty} \mathbf{u}_k$. Then, \mathbf{u}_∞ is a KKT point for the plant optimization problem (3.4a) subject to (3.4b).*

Proof. Stacking the inequality constraints and using the definition of the functions $\phi(\cdot)$ and

$\mathbf{g}(\cdot)$, the equivalence of (3.20) with the standard MA scheme (2.4) follows immediately. Hence, Theorem 2.2 applies and gives the desired result. \square

In summary, the distributed MA Algorithm 3.3 requires the least modeling effort but relies the most on (if possible fairly accurate) measurements.

3.4 Numerical Example

To illustrate how the two proposed distributed modifier-adaptation schemes work, they are applied to a simple numerical example from the literature, which does not require any domain-specific knowledge.

3.4.1 Plant Description

The plant consisting of two interconnected subsystems as depicted in Figure 3.6 (Brdyś and Tatjewski, 2005) is studied.

The interconnection structure of the system is reflected by the equalities

$$v_{p,1} - y_{p,21} = 0, \quad (3.23a)$$

$$v_{p,2} - y_{p,1} = 0, \quad (3.23b)$$

and the steady-state outputs of the subsystems are:

$$y_{p,11} = 4u_1 + v_{p,1}^2, \quad (3.24a)$$

$$y_{p,21} = 0.2e^{u_2} - 0.1u_2 v_{p,2}, \quad (3.24b)$$

$$y_{p,22} = 0.1u_2 v_{p,2} - 0.2u_2^2. \quad (3.24c)$$

$$(3.24d)$$

The objective is to minimize the sum of the two costs

$$\Phi_{p,1} = -6u_1^2 - 5u_1 v_{p,1} + (y_{p,11} - 1)^2, \quad (3.25a)$$

$$\Phi_{p,2} = u_2^2 + 4u_2 + 0.5(y_{p,22})^2. \quad (3.25b)$$

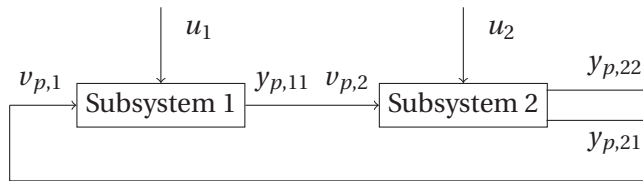


Figure 3.6 – Structure of the plant with two interconnected subsystems.

subject to the box constraints

$$G_{p,1}(u_1) = u_1 - 0.2 \leq 0, \quad (3.26a)$$

$$G_{p,2}(u_2) = u_2 - 3 \leq 0. \quad (3.26b)$$

The steady-state inputs that optimize the plant performance are $\mathbf{u}_p^* = (0.2; -2.019)^\top$.

3.4.2 Available Model

The available output model is as follows:

$$y_{11} = -u_1 + 0.6v_1, \quad (3.27a)$$

$$y_{21} = 0.63u_2 + 0.5v_2, \quad (3.27b)$$

$$y_{22} = 2.4u_2 + 2.2v_2, \quad (3.27c)$$

$$(3.27d)$$

The model cost functions are given as,

$$\Phi_1 = u_1^2 + 6u_1 + (y_{11} - 1)^2, \quad (3.28a)$$

$$\Phi_2 = u_2^2 + 4u_2 + 0.5(y_{22})^2. \quad (3.28b)$$

the model constraints are different:

$$G_1(u_1) = u_1 - 0.1 \leq 0, \quad (3.29a)$$

$$G_2(u_2) = u_2 - 3.5 \leq 0. \quad (3.29b)$$

Note that these optimal plant inputs are outside the feasible region of the model, that is, they are infeasible for the nominal model-based optimization problem (3.27)–(3.29). As will be seen in the following, this will not be a concern for the proposed distributed modifier-adaptation algorithms. Furthermore, for all methods, exact plant gradients and noise-free measurements are assumed to be available.

3.4.3 Simulation Results

The problem described in the previous section is solved via Algorithms 3.2, and 3.3. Each of them is initialized at $\mathbf{u}(0) = (-0.5, 0)$. The plant has two inputs u_1 and u_2 , thus the filter matrix \mathbf{K} is 2×2 diagonal matrix. For both algorithms, a diagonal filter matrix of the form $\mathbf{K} = \text{diag}([0.15 \ 0.3])$ is chosen. This choice of low filter gains indicate that both algorithms have to take very cautious steps to avoid oscillations and divergence. In particular, the diagonal values larger than 0.35 lead to divergence of Algorithm 3.3, thus being more sensitive to the increase of eigenvalues of \mathbf{K} in comparison to Algorithms 3.2 for this particular example.

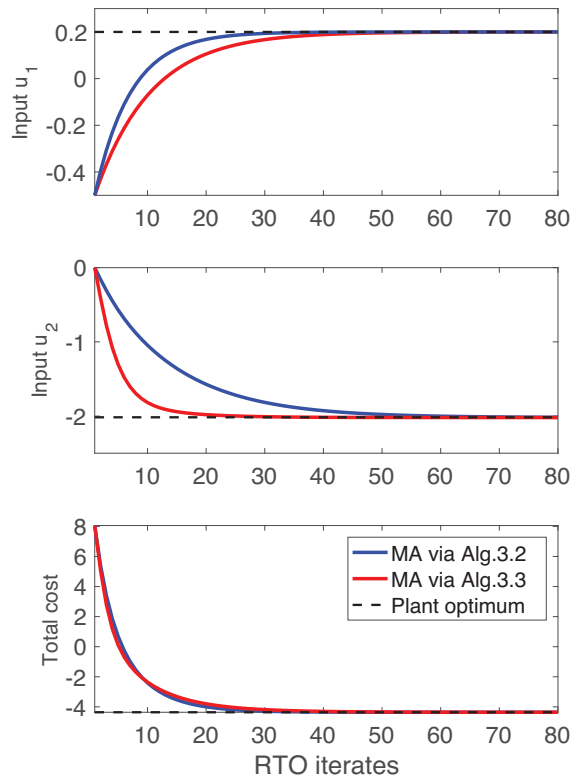


Figure 3.7 – Convergence of the input sequences and the plant cost.

As expected, the absence of accurate interconnection model, or of both the interconnection and local output models, leads to slow convergence for Algorithms 3.2 and 3.3, respectively. Figure 3.7 shows that the inputs converge only after about 40 to 50 RTO iterations, whereas the plant cost converge to optimal value after about around 30 iterations. Still, the algorithms take too many plant setpoint changes knowing that Figure 3.7 depicts only RTO iterations without additional perturbations required for gradient estimation.

3.5 Conclusion

This chapter has discussed two distributed modifier-adaptation schemes for the real-time optimization of interconnected systems. Such systems are increasingly encountered in many areas of engineering, and distributed approaches to optimize the steady-state performance in the presence of plant-model mismatch are highly relevant. The schemes proposed here accomplish this task, and offer additional benefits such as confidentiality of subsystem models. The method of choice depends on a variety of factors, including the quality of the models and of the measurements and the confidentiality of the models.

On the one hand, Algorithms 3.2 and 3.3, demonstrate the strength of the modifier-adaptation methodology even when there is no model of the subsystem interconnections. The model used does not assume interaction between the subsystems, and the price to pay for this includes more RTO iterations for convergence to the plant optimum.

On the other hand, one may face severe practical difficulties in the application of these two methods to a particular system. The problem of estimating the plant gradients is the most challenging part in modifier-adaptation schemes. The presented algorithms require the knowledge of the sensitivities of each subsystem's cost and constraint functions with respect to the inputs of other subsystems. In practice, obtaining these estimates can be very difficult depending on the dimensionality of the system, the dynamics of the whole plant and the time it takes to go to steady state. Moreover, the complexity of estimating these plant gradients would be comparable to that of treating the whole plant as one unit. Furthermore, the assumptions regarding the inequalities in Section 3.3.1 may be rather restrictive since they do not allow the inequality constraints to depend on the interconnection variables.

4 Modifier Adaptation for Interconnected Systems using Local Modifiers

This chapter is based on:

P. Milosavljevic, R. Schneider, T. Faulwasser, and D. Bonvin. Distributed modifier adaptation using a coordinator and measured interconnection variables. In 20th IFACWorld Congress, Toulouse, France, 2017.

P. Milosavljevic, R. Schneider, A. Cortinovis, T. Faulwasser, and D. Bonvin. A distributed feasible-side convergentmodifier-adaptation scheme for interconnected systems, with application to gas-compressor stations. *Comp. Chem. Eng.*, 115:474–486, 2018c.

This chapter focuses on model-based steady-state optimization of interconnected systems utilizing an interconnection model and local measurements. As a continuation of the previous chapter, we deal here with plants that consist of interconnected subsystems, which implies that the subsystems are influenced not only by their decision variables (inputs) but also by interconnection variables that are outputs of other subsystems. Hence, these interconnection variables cannot be directly manipulated. We assume that the measurements of the interconnection variables \mathbf{v} and outputs \mathbf{y} are available. In fact, this chapter extends the MA framework for interconnected systems presented in the previous chapter to the schemes that use only local measurements of each subsystem for plant gradient estimates. Namely, the proposed schemes rely on the use of *local modifiers*. This means that the gradients of i -th subsystem with respect to its own local inputs \mathbf{u}_i and \mathbf{v}_i are required. Furthermore, we propose two MA formulations.

Firstly, a centralized MA scheme is proposed and its KKT matching properties are analyzed. The second scheme is a coordinator-based distributed MA scheme. The proposed algorithm has a two-layer structure. In the inner loop, a coordinator ensures that the inputs of the local subproblems are consistent with the interconnection model. Upon convergence of the inner loop, these consistent inputs are applied to the plant in the outer loop. Note that, the proposed scheme does not require the models underlying each subproblems to be

shared with other subsystems for purpose of obtaining the plant gradient estimates. This is possible due to a novel way of updating the modifiers in the distributed RTO framework. Furthermore, we propose a distributed modifier-adaptation algorithm that, besides the interconnection model, employs a coordinator, and we prove its feasible-side convergence to the plant optimum.

Section 4.1 gives the description of the plant and model optimization problem. Section 4.2 presents the centralized MA problem of optimizing an uncertain plant with interconnected subsystems. In Section 4.3, the distributed MA algorithm is proposed and analyzed. We describe these two algorithms, namely a centralized MA and a decentralized MA, both using measurement of interconnection variables. Their effectiveness is illustrated in Section 4.5 on a numerical example and a load-sharing optimization problem of serial compressors. Finally, Section 4.6 concludes the chapter.

4.1 Problem Formulation

Here we present the plant and model optimization problem with the same structure as given in Chapter 4. Since outputs are functions of \mathbf{u} and \mathbf{v} , for the sake of simplicity, we exclude cost and inequality constraint dependence on outputs and write them directly as functions of \mathbf{u} and \mathbf{v} .

4.1.1 Plant Optimization Problem

The problem of finding optimal operating conditions for a plant with interconnected systems and known interconnection model, depicted in Figure 4.1, can be formulated as the following NLP

$$(\mathbf{u}_p^*, \mathbf{v}_p^*) = \underset{\mathbf{u}, \mathbf{v}_p}{\operatorname{argmin}} \Phi_p(\mathbf{u}, \mathbf{v}_p) \quad (4.1a)$$

$$\text{s.t. } \mathbf{y}_p := \mathbf{F}_p(\mathbf{u}, \mathbf{v}_p) \quad (4.1b)$$

$$\mathbf{G}_p(\mathbf{u}, \mathbf{v}_p) \leq \mathbf{0} \quad (4.1c)$$

$$\mathbf{v}_p = \mathbf{H}_p(\mathbf{y}_p) \quad (4.1d)$$

where $\mathbf{u} \in \mathbb{R}^{n_u}$, $\mathbf{v}_p \in \mathbb{R}^{n_v}$ and $\mathbf{y}_p \in \mathbb{R}^{n_y}$ are the steady-state plant inputs, interconnection variables and outputs, respectively. $\Phi_p : \mathbb{R}^{n_u} \times \mathbb{R}^{n_v} \rightarrow \mathbb{R}$ is the cost of the whole plant, $\mathbf{G}_p : \mathbb{R}^{n_u} \times \mathbb{R}^{n_v} \rightarrow \mathbb{R}^{n_G}$ are the process constraints and $\mathbf{H}_p \in \mathbb{R}^{n_v}$ is the interconnection function. We consider that the plant consists of N interconnected subsystems, $i \in \mathcal{N} = \{1, \dots, N\}$. Subsystem i has the inputs $\mathbf{u}_i \in \mathbb{R}^{n_{u_i}}$, the interconnection variables $\mathbf{v}_{p,i} \in \mathbb{R}^{n_{v_i}}$, the outputs $\mathbf{y}_{p,i} \in \mathbb{R}^{n_{y_i}}$ are defined as in (3.8c)-(3.8f) while the constraints $\mathbf{G}_{p,i} \in \mathbb{R}^{n_{G_i}}$ are organized as

$$\mathbf{G}_p(\mathbf{u}, \mathbf{v}_p) = \operatorname{col}(\mathbf{G}_{p,1}(\mathbf{u}_1, \mathbf{v}_{p,1}), \dots, \mathbf{G}_{p,N}(\mathbf{u}_N, \mathbf{v}_{p,N})). \quad (4.2a)$$

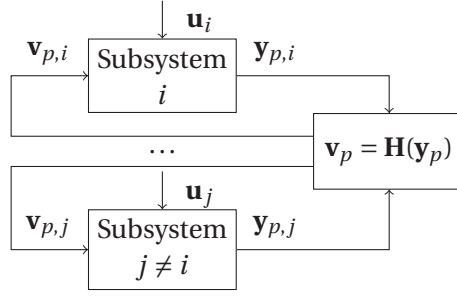


Figure 4.1 – Plant composed of interconnected subsystems. The interconnection variables are the inputs \mathbf{v}_p .

Furthermore, we assume that the objective Φ_p is separable,

$$\Phi_p(\mathbf{u}, \mathbf{v}) = \sum_{i \in \mathcal{N}} \Phi_{p,i}(\mathbf{u}_i, \mathbf{v}_{p,i}). \quad (4.2b)$$

4.1.2 Model Optimization Problem

In practice, the plant functions $\Phi_p, \mathbf{F}_p, \mathbf{G}_p, \mathbf{H}_p$ are usually not known accurately; only models of these functions are available. Hence, the structure shown in Figure 4.1 is also assumed for the model, with which *only an approximate solution* to Problem (4.1) can be obtained by solving the following model-based problem:

$$(\mathbf{u}^*, \mathbf{v}^*) = \underset{\mathbf{u}, \mathbf{v}}{\operatorname{argmin}} \Phi(\mathbf{u}, \mathbf{v}) \quad (4.3a)$$

$$\text{s.t. } \mathbf{y} := \mathbf{F}(\mathbf{u}, \mathbf{v}) \quad (4.3b)$$

$$\mathbf{G}(\mathbf{u}, \mathbf{v}) \leq \mathbf{0} \quad (4.3c)$$

$$\mathbf{v} = \mathbf{H}(\mathbf{y}), \quad (4.3d)$$

where $\mathbf{v} \in \mathbb{R}^{n_v}$ and $\mathbf{y} \in \mathbb{R}^{n_y}$ are the modeled interconnection variables and outputs, respectively. Similar to the plant, the constraint models \mathbf{G} , and the cost function Φ are defined as:

$$\mathbf{G}(\mathbf{u}, \mathbf{v}) = \operatorname{col}(\mathbf{G}_1(\mathbf{u}_1, \mathbf{v}_1), \dots, \mathbf{G}_N(\mathbf{u}_N, \mathbf{v}_N)), \quad (4.4a)$$

$$\Phi(\mathbf{u}, \mathbf{v}) = \sum_{i \in \mathcal{N}} \Phi_i(\mathbf{u}_i, \mathbf{v}_i). \quad (4.4b)$$

4.2 Centralized MA with Interconnection Model

The process measurements are used to iteratively modify the model-based problem (4.3) by introducing local modifiers. Namely, a centralized formulation of the scheme is proposed,

given by the following sequence of optimization problems

$$\left(\mathbf{u}_{k+1}^*, \mathbf{v}_{k+1}^* \right) = \underset{\mathbf{u}, \mathbf{v}}{\operatorname{argmin}} \Phi(\mathbf{u}, \mathbf{v}) + (\boldsymbol{\lambda}_k^\Phi)^\top (\mathbf{z} - \mathbf{z}_{p,k}) \quad (4.5a)$$

$$\text{s.t. } \mathbf{y} := \mathbf{F}(\mathbf{u}, \mathbf{v}) \quad (4.5b)$$

$$\mathbf{G}_m := \mathbf{G}(\mathbf{u}, \mathbf{v}) + \boldsymbol{\varepsilon}_k^G + (\boldsymbol{\lambda}_k^G)^\top (\mathbf{z} - \mathbf{z}_{p,k}) \leq 0 \quad (4.5c)$$

$$\mathbf{v} = \mathbf{H}(\mathbf{y}) + \boldsymbol{\varepsilon}_k^H + (\boldsymbol{\lambda}_k^H)^\top (\mathbf{z} - \mathbf{z}_{p,k}), \quad (4.5d)$$

where $\mathbf{z}^\top := [\mathbf{z}_1^\top, \dots, \mathbf{z}_N^\top]$ and $\mathbf{z}_p^\top := [\mathbf{z}_{p,1}^\top, \dots, \mathbf{z}_{p,N}^\top]$ with, for all $i \in \mathcal{N}$, $\mathbf{z}_i^\top := [\mathbf{u}_i^\top, \mathbf{v}_i^\top]$ and $\mathbf{z}_{p,i}^\top := [\mathbf{u}_i^\top, \mathbf{v}_{p,i}^\top]$. Furthermore, $\boldsymbol{\lambda}_k^\Phi \in \mathbb{R}^{n_u+n_v}$ is the vector of first-order cost modifiers, $\boldsymbol{\lambda}_k^H \in \mathbb{R}^{(n_u+n_v) \times n_Q}$ are first-order modifiers of interconnection constraints, $\boldsymbol{\varepsilon}_k^H \in \mathbb{R}^{n_Q}$ are zero-order modifiers of interconnection constraints, $\boldsymbol{\lambda}_k^G \in \mathbb{R}^{(n_u+n_v) \times n_G}$ are first-order constraint modifiers, and $\boldsymbol{\varepsilon}_k^G \in \mathbb{R}^{n_G}$ are zero-order constraint modifiers. These modifiers are defined as:

$$\boldsymbol{\lambda}_k^\Phi = \nabla_{\mathbf{z}_p} \Phi_p(\mathbf{u}_k, \mathbf{v}_{p,k}) - \nabla_{\mathbf{z}} \Phi(\mathbf{u}_k, \mathbf{v}_{p,k}) \quad (4.6a)$$

$$\boldsymbol{\varepsilon}_k^G = \mathbf{G}_p(\mathbf{u}_k, \mathbf{v}_{p,k}) - \mathbf{G}(\mathbf{u}_k, \mathbf{v}_{p,k}) \quad (4.6b)$$

$$\boldsymbol{\lambda}_k^G = \nabla_{\mathbf{z}_p} \mathbf{G}_p(\mathbf{u}_k, \mathbf{v}_{p,k}) - \nabla_{\mathbf{z}} \mathbf{G}(\mathbf{u}_k, \mathbf{v}_{p,k}) \quad (4.6c)$$

$$\boldsymbol{\varepsilon}_k^H = \mathbf{H}_p(\mathbf{y}_p(\mathbf{u}_k, \mathbf{v}_{p,k})) - \mathbf{H}(\mathbf{y}(\mathbf{u}_k, \mathbf{v}_{p,k})) \quad (4.6d)$$

$$\boldsymbol{\lambda}_k^H = \nabla_{\mathbf{z}_p} \mathbf{H}_p(\mathbf{y}_p(\mathbf{u}_k, \mathbf{v}_{p,k})) - \nabla_{\mathbf{z}} \mathbf{H}(\mathbf{y}(\mathbf{u}_k, \mathbf{v}_{p,k})). \quad (4.6e)$$

Moreover, we assume that the new inputs are filtered before being applied to the plant,

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \mathbf{K}(\mathbf{u}_{k+1}^* - \mathbf{u}_k). \quad (4.7)$$

In its simplest form, the algorithm proceeds as follows

Algorithm 4.1 : Centralised MA with Local Modifiers

Initialization: choose the diagonal filter matrix \mathbf{K} with eigenvalues in the interval $(0, 1]$ and choose a feasible input vector \mathbf{u}_0 .

for $k = 0 \rightarrow \infty$ **do:**

1. Apply the inputs \mathbf{u}_k to the plant and wait for steady state.
2. Update the modifiers according to (4.6a)-(4.6e).
3. Solve the modified model-based optimization problem (4.5)
4. Filter the inputs according to (4.7).

end

Assumption 4.1 (Existence of a solution).

We assume that the solution to Problem (4.5) exists at every iteration k and that it is a regular KKT point. \square

Assumption 4.2 (Unique solution).

For every admissible input vector \mathbf{u} , there is a unique vector \mathbf{v} such that the pair (\mathbf{u}, \mathbf{v}) is solution to equations (4.5d). \square

Now we show that fixed point of Algorithm 4.1 is a KKT point of the plant problem (4.1).

Proposition 4.1 (KKT matching for Algorithm 4.1).

Let Assumptions 4.1–4.2 hold and let Algorithm 4.1 converge to $(\mathbf{u}_\infty, \mathbf{v}_\infty)$. Then $(\mathbf{u}_\infty, \mathbf{v}_\infty)$ is a KKT point of Problem (4.1). \square

Proof. The proof can be found in Appendix C.4. \square

Remark 4.1 (Alternative modifier formulation).

Note that the modifiers (4.6a)–(4.6e), estimated at point $(\mathbf{u}, \mathbf{v}_p)$, are written with respect to the inputs \mathbf{u} and the interconnection variables \mathbf{v} . Also, note that the plant variables \mathbf{v}_p differ from \mathbf{v} due to model uncertainty. Hence, the modifiers differ from those in standard schemes, where only gradient information with respect to the decision variables \mathbf{u} is used (Marchetti et al., 2009, 2016). \square

Note that outputs \mathbf{y} are not the decision variables in Problem 4.5. Here, outputs are used to define \mathbf{v} in (4.5d) but they will play an important role in distributed schemes as we will see next. In the following section, the structure of Problem (4.5) is exploited for the distributed implementation of MA scheme.

4.3 Distributed MA with Interconnection Model

Here, the distributed MA scheme for interconnected systems with the plant and model structure in Figure 4.1 is proposed. Also, a feasible-side convergent distributed MA scheme is investigated.

Prior to details on MA schemes, the following assumption regarding the interconnection constraint in (4.1d) and (4.3d) is stated.

Assumption 4.3 (Interconnections are linear in outputs).

The interconnection functions $\mathbf{H}_p(\mathbf{y}_p)$ and $\mathbf{H}(\mathbf{y})$ are linear in outputs \mathbf{y}_p and \mathbf{y} , respectively, namely, $\mathbf{H}_p(\mathbf{y}_p) = \mathbf{H}_p \mathbf{y}_p$ and $\mathbf{H}(\mathbf{y}) = \mathbf{H} \mathbf{y}$, where the matrix \mathbf{H} is known. Furthermore, \mathbf{H} is assumed to consist of zero and ones only, with precisely one non-zero element in each row and each column. \square

Note that assuming knowledge of the interconnection matrix \mathbf{H} is realistic for physical systems where the outputs of one unit are inputs to another unit. A real-world example satisfying Assumption 4.3 will be discussed for the problem of scheduling gas compressor stations.

4.3.1 Distributed MA Formulation

As the cost and constraints of all subproblems are functions of only the local variables \mathbf{u}_i and \mathbf{v}_i , we now reformulate optimization problem such that the structure is suitable for distributed algorithms. To this end, consider

$$\left(\mathbf{u}_{k+1}^*, \mathbf{v}_{k+1}^* \right) = \underset{\mathbf{u}, \mathbf{v}}{\operatorname{argmin}} \sum_{i \in \mathcal{N}} \left(\Phi_i(\mathbf{u}_i, \mathbf{v}_i) + \left(\boldsymbol{\lambda}_k^{\Phi_i} \right)^\top (\mathbf{z}_i - \mathbf{z}_{p,i,k}) \right) \quad (4.8a)$$

$$\text{s.t. } \forall i \in \mathcal{N} :$$

$$\mathbf{y}_{m,i} := \mathbf{F}_i(\mathbf{u}_i, \mathbf{v}_i) + \boldsymbol{\varepsilon}_k^{F_i} + \left(\boldsymbol{\lambda}_k^{F_i} \right)^\top (\mathbf{z}_i - \mathbf{z}_{p,i,k}) \quad (4.8b)$$

$$\mathbf{G}_i(\mathbf{u}_i, \mathbf{v}_i) + \boldsymbol{\varepsilon}_k^{G_i} + \left(\boldsymbol{\lambda}_k^{G_i} \right)^\top (\mathbf{z}_i - \mathbf{z}_{p,i,k}) \leq 0 \quad (4.8c)$$

$$\mathbf{v} = \mathbf{H}\mathbf{y}_m, \quad (4.8d)$$

where $\boldsymbol{\lambda}_k^{\Phi_i} \in \mathbb{R}^{n_{u_i} + n_{v_i}}$, $\boldsymbol{\lambda}_k^{F_i} \in \mathbb{R}^{(n_{u_i} + n_{v_i}) \times n_{y_i}}$, $\boldsymbol{\lambda}_k^{G_i} \in \mathbb{R}^{(n_{u_i} + n_{v_i}) \times n_{G_i}}$ are the first-order modifiers and $\boldsymbol{\varepsilon}_k^{F_i} \in \mathbb{R}^{n_{y_i}}$, $\boldsymbol{\varepsilon}_k^{G_i} \in \mathbb{R}^{n_{G_i}}$ are the zero-order modifiers; n_{G_i} is the number of inequality constraints and n_{y_i} is the number of measured outputs for each i -th subsystem. The modifiers are obtained as:

$$\boldsymbol{\lambda}_k^{\Phi_i} = \nabla_{\mathbf{z}_{p,i}} \Phi_{p,i}(\mathbf{u}_{i,k}, \mathbf{v}_{p,i,k}) - \nabla_{\mathbf{z}_i} \Phi_i(\mathbf{u}_{i,k}, \mathbf{v}_{p,i,k}) \quad (4.8e)$$

$$\boldsymbol{\varepsilon}_k^{G_i} = \mathbf{G}_{p,i}(\mathbf{u}_{i,k}, \mathbf{v}_{p,i,k}) - \mathbf{G}_i(\mathbf{u}_{i,k}, \mathbf{v}_{p,i,k}) \quad (4.8f)$$

$$\boldsymbol{\lambda}_k^{G_i} = \nabla_{\mathbf{z}_{p,i}} \mathbf{G}_{p,i}(\mathbf{u}_{i,k}, \mathbf{v}_{p,i,k}) - \nabla_{\mathbf{z}_i} \mathbf{G}_i(\mathbf{u}_{i,k}, \mathbf{v}_{p,i,k}) \quad (4.8g)$$

$$\boldsymbol{\varepsilon}_k^{F_i} = \mathbf{F}_{p,i}(\mathbf{u}_{i,k}, \mathbf{v}_{p,i,k}) - \mathbf{F}_i(\mathbf{u}_{i,k}, \mathbf{v}_{p,i,k}) \quad (4.8h)$$

$$\boldsymbol{\lambda}_k^{F_i} = \nabla_{\mathbf{z}_{p,i}} \mathbf{F}_{p,i}(\mathbf{u}_{i,k}, \mathbf{v}_{p,i,k}) - \nabla_{\mathbf{z}_i} \mathbf{F}_i(\mathbf{u}_{i,k}, \mathbf{v}_{p,i,k}). \quad (4.8i)$$

Prior to discussion about the properties of the scheme, the following assumption regarding the model is stated.

Assumption 4.4 (Convex nominal models).

In Problem 4.3, the nominal model has strictly convex cost functions Φ_i , linear inequality functions \mathbf{G}_i and linear steady-state outputs \mathbf{F}_i , $\forall i \in \mathcal{N}$. \square

Problem (4.8) has a special structure that can be exploited for its distributed implementation. In particular, at every iteration k of Problem (4.8), the cost function (4.8a), the equality constraints (4.8b) and the inequality constraints (4.8c) are functions of the local variables \mathbf{z}_i . The coupled equality constraints (4.8d) can be handled by moving them into the objective function upon formulating the augmented Lagrangian. Since the modified cost function (4.8a) is convex (due to Assumption 4.4) and the constraints (4.8b)-(4.8d) are linear, dual decomposition has been proposed to obtain a solution in distributed way (Necoara et al., 2011). In particular, Problem (4.8) can be solved iteratively with a two-step procedure:

Step 1: For all $i \in \mathcal{N}$, obtain the solution to the following optimization problem:

$$\begin{aligned} (\mathbf{u}_{i,k+1}^{l,*}, \mathbf{v}_{i,k+1}^{l,*}) = \underset{(\mathbf{u}_i, \mathbf{v}_i) \in \mathbf{G}_i}{\operatorname{argmin}} & \Phi_i(\mathbf{u}_i, \mathbf{v}_i) + \\ & + \left(\boldsymbol{\lambda}_k^{\Phi_i} \right)^\top (\mathbf{z}_i - \mathbf{z}_{p,i,k}) + \left(\boldsymbol{\mu}_i^l \right)^\top \mathbf{v}_i - \sum_{j \in \mathcal{N}} \left(\boldsymbol{\mu}_j^l \right)^\top \mathbf{H}_{ij} \mathbf{y}_{m,i} \end{aligned} \quad (4.9)$$

Step 2: Update of dual variable $\boldsymbol{\mu}$:

$$\mathbf{y}_{m,i,k+1}^{l,*} := \mathbf{F}_i(\mathbf{u}_{i,k+1}^{l,*}, \mathbf{v}_{i,k+1}^{l,*}) + \boldsymbol{\varepsilon}_k^{F_i} + \left(\boldsymbol{\lambda}_k^{F_i} \right)^\top \left(\mathbf{z}_{i,k+1}^{l,*} - \mathbf{z}_{p,i,k} \right), \forall i \in \mathcal{N} \quad (4.10a)$$

$$\boldsymbol{\mu}^{l+1} = \boldsymbol{\mu}^l + \rho_\mu \left(\mathbf{v}_{k+1}^{l,*} - \mathbf{H} \mathbf{y}_{m,k+1}^{l,*} \right) \quad (4.10b)$$

where $\mathbf{H}_{ij} \in \mathbb{R}^{n_{v_j} \times n_{y_i}}$, $\boldsymbol{\mu}$ is the dual variable of the equality constraint (4.8d), $\rho_\mu > 0$, and l is the counter of the model-based inner-loop iterations. As a result, we propose Algorithm 4.2 as a distributed MA scheme that exploits the knowledge of the interconnection model. As can be seen in Figure 4.2, Algorithm 4.2 required existence of the coordinator to ensure meeting the coupled equality constraints. In its simplest form, the algorithm proceeds as follows

Algorithm 4.2 : Distributed MA with Local Modifiers

Initialize: Choose a step size parameter ρ_μ , an initial dual variable $\boldsymbol{\mu}^0$, a sequence of filter gain matrices \mathbf{K}_k for all nonnegative integers k , and a feasible input vector \mathbf{u}_0 .

for $k = 0 \rightarrow \infty$ **do:**

1. Apply the inputs \mathbf{u}_k to the plant and wait for steady state.
2. Measure the interconnection variables \mathbf{v}_p . Also, measure the plant cost, constraints and outputs, and estimate their gradients with respect to \mathbf{z}_p .
3. Update the modifiers according to (4.8e)-(4.8i).
4. **for** $l = 0 \rightarrow$ convergence **do:**
 - (a) For all $i \in \mathcal{N}$, compute in parallel the solutions $(\mathbf{u}_{i,k+1}^{l,*}, \mathbf{v}_{i,k+1}^{l,*})$ to the optimization problem (4.9).
 - (b) Update the dual variable $\boldsymbol{\mu}$ according to (4.10).

end

5. Filter the inputs according to (4.7).

end

Now, we propose the following

Proposition 4.2 (KKT matching for Algorithm 4.2).

Let Assumptions 4.1–4.4 hold. Then,

- (i) there exist a $\rho_\mu > 0$ such that the inner loop of Algorithm 4.2, defined by (4.9)–(4.10), converges to the optimum of Problem (4.8).
- (ii) If in addition, in the outer loop, the modifiers in (4.8a)–(4.8d) are updated according to (4.8e)–(4.8i), then, upon convergence of the outer loop, a KKT point $(\mathbf{u}_p^*, \mathbf{v}_p^*)$ of the plant Problem (4.1) is reached.

□

Proof. Part (i): The iterative procedure of Steps (4.9) and (4.10) is essentially a dual decomposition discussed in (Necoara et al., 2011, Thm. 3.5). Furthermore, due to Assumption 4.4, Problem (4.8) is strictly convex. Hence, convergence of the inner loop (4.9)–(4.10) to the optimal solution of its centralized formulation (4.8) can be enforced via a proper choice of the tuning parameter ρ_μ .

Part (ii): Upon convergence of the outer loop, optimal values $(\mathbf{u}_\infty^*, \mathbf{v}_\infty^*)$ of Problem (4.8) correspond to the KKT point of plant problem (4.1). This can be shown in the similar way as in Proposition 4.1. □

Remark 4.2 (Local modifier estimation).

Note that each subproblem i relies on the usage of the local input-output measurements u_i and $v_{p,i}$ to handle uncertainty locally. Due to the linearity of the constraints (Assumption 4.4), there is no need to exchange the plant and model gradient information among subsystems. □

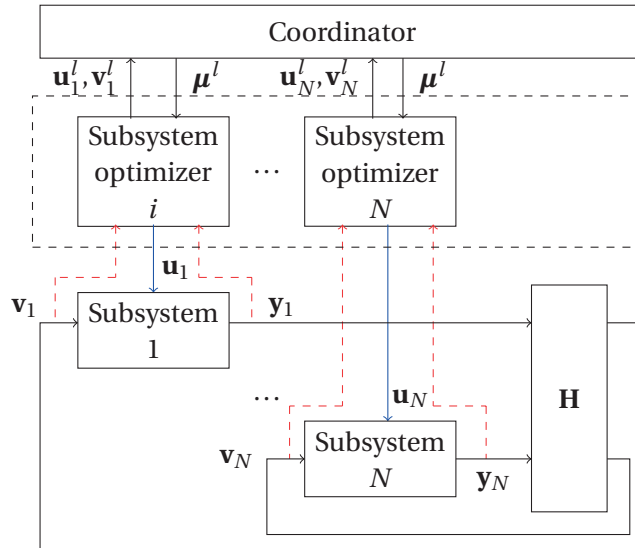


Figure 4.2 – Coordination-based structure of Algorithm 4.2.

Remark 4.3 (Advantages of using convex models).

The use of convex models guarantees model adequacy without prior knowledge of the plant optimum (Marchetti et al., 2009; François and Bonvin, 2013a). Furthermore, adding affine modifiers to the cost and constraints will preserve a convex structure to MA optimization problem. This implies that distributed convex solvers can be used, whose convergence properties are well established (Boyd et al., 2011). However, more RTO iterations might be required if the convex models are not good plant approximations. \square

Remark 4.4 (Non-convex models).

Note that assumption regarding convexity is required only for convergence of inner-loop iterations. These iterations are model-based and one may choose different algorithm to obtain the solution of (4.9)-(4.10) depending on the cost and constraint structure of Problem (4.3). For example, if Problem (4.3) is non-convex, one might choose techniques detailed by Hours and Jones (2016); Houska et al. (2016). \square

4.3.2 A Feasible-Side Convergent Distributed MA Algorithm

The distributed MA methodology proposed in Subsection 4.3.1 guarantees optimality of the plant upon convergence. In addition, the design of a distributed method should enforce properties such as the feasibility of the plant constraints (4.1c) as well as monotonic improvement of the plant cost through all RTO iterates. Here, a distributed MA algorithm that guarantees feasible-side convergence to the plant optimum is proposed as a modification of the distributed algorithm presented in Subsection 4.3.1.

Firstly, we define the cost functions and constraint functions by adding second-order terms for i -th subproblem:

$$\Phi_{i,k}^U(\mathbf{u}_i, \mathbf{v}_i) := \Phi_i(\mathbf{u}_i, \mathbf{v}_i) + \left(\boldsymbol{\lambda}_k^{\Phi_i}\right)^\top (\mathbf{z}_i - \mathbf{z}_{p,i,k}) + \frac{\delta^{\Phi_i}}{2} \|\mathbf{z}_i - \mathbf{z}_{p,i,k}\|_2^2, \quad (4.11)$$

$$G_{i,j,k}^U(\mathbf{u}_i, \mathbf{v}_i) := G_{i,j}(\mathbf{u}_i, \mathbf{v}_i) + \boldsymbol{\varepsilon}_k^{G_{i,j}} + \left(\boldsymbol{\lambda}_k^{G_{i,j}}\right)^\top (\mathbf{z}_i - \mathbf{z}_{p,i,k}) + \frac{\delta^{G_{i,j}}}{2} \|\mathbf{z}_i - \mathbf{z}_{p,i,k}\|_2^2 \leq 0, \quad j = 1, \dots, n_{G_i}. \quad (4.12)$$

where $\delta^{\Phi_i}, \delta^{G_{i,j}}$ are some positive scalars.

The feasible convex set for i -th subsystem at the k -th RTO iteration is defined as

$$\mathcal{F}_{i,k}^U = \{(\mathbf{u}_i, \mathbf{v}_i) \in \mathbb{R}^{u_i} \times \mathbb{R}^{v_i} : G_{i,j,k}^U(\mathbf{u}_i, \mathbf{v}_i) \leq 0, j = 1, \dots, n_{G_i}\}.$$

Let us now propose the following distributed MA algorithm:

Chapter 4. Modifier Adaptation for Interconnected Systems using Local Modifiers

Algorithm 4.3 : Distributed Feasible-Side Convergent MA with Local Modifiers

1. *Initialization*: Choose a step size parameter ρ_μ , an initial dual variable $\boldsymbol{\mu}^0$ and a feasible input vector \mathbf{u}_0 , and set $k := 0$.
2. *Plant experiment*: Apply the inputs \mathbf{u}_k to the plant and wait for steady state.
3. *Modifier computation*: Compute the modifiers as per (4.8e)-(4.8i).

4. *New input calculation*:

for $l = 0 \rightarrow$ convergence **do**:

(a) For all $i \in \mathcal{N}$, compute in parallel the solutions

$$\left(\mathbf{u}_{i,k+1}^{l,*}, \mathbf{v}_{i,k+1}^{l,*} \right) = \underset{(\mathbf{u}_i, \mathbf{v}_i) \in \mathcal{F}_{i,k}^U}{\operatorname{argmin}} \Phi_{i,k}^U(\mathbf{u}_i, \mathbf{v}_i) + \left(\boldsymbol{\mu}_i^l \right)^\top \mathbf{v}_i - \sum_{j \in \mathcal{N}} \boldsymbol{\mu}_j^{l,\top} \mathbf{H}_{ij} \mathbf{y}_{m,i} \quad (4.13)$$

(b) Update the dual variable $\boldsymbol{\mu}$ according to (4.10).

end

5. *Iterate*: Set $k := k + 1$ and return to Step 2.

To establish convergence of this algorithm, we need that Assumption 3.1 holds. This means that interconnection variables can be expressed as the functions of input vector \mathbf{u} . This assumption allows transforming the plant functions with interconnection variables \mathbf{v} into RTO formulation where cost and constraints are functions of vector \mathbf{u}

$$\mathbf{h}_p = \operatorname{col}(\mathbf{h}_{p,1}, \dots, \mathbf{h}_{p,N}), \quad (4.14a)$$

$$\mathbf{g}_{p,i,j}(\mathbf{u}) = G_{p,i,j}(\mathbf{u}_i, \mathbf{h}_{p,i}(\mathbf{u})), \quad j = 1, \dots, n_{G_i} \quad (4.14b)$$

$$\phi_{p,i}(\mathbf{u}) = \Phi_{p,i}(\mathbf{u}_i, \mathbf{h}_{p,i}(\mathbf{u})), \quad \forall i \in \mathcal{N}, \quad (4.14c)$$

$$\phi_p(\mathbf{u}) = \sum_{i \in \mathcal{N}} \phi_{p,i}(\mathbf{u}). \quad (4.14d)$$

Assumption 4.5 (Differentiable functions).

Let $\Phi_i(\mathbf{u}_i, \mathbf{v}_i)$, $\phi_p(\mathbf{u})$ and $\mathbf{g}_{p,i,j}(\mathbf{u})$ be differentiable functions and let their gradients be $\delta_i^{\Phi_m}$, δ^{ϕ_p} and $\delta_{i,j}^{\mathbf{g}_p}$ -Lipschitz continuous, respectively. \square

Assumption 4.6 (Invertibility).

The matrix $\left(\mathbf{I} - \mathbf{H} \nabla_{\mathbf{v}_p} \mathbf{F}_p^\top(\mathbf{z}_{p,k}) \right)$, where the plant functions \mathbf{F}_p are used in (4.1b) and \mathbf{H} in Assumption 4.3, is invertible. \square

Adding second-order terms in (4.11) forces monotonic decrease of the plant cost in the distributed MA algorithm as will be show next. Hence, the properties of Algorithm 4.3 are stated in the following proposition:

Proposition 4.3 (Feasible-side convergence).

Let Assumptions 3.1, and 4.3–4.6 hold. Then,

4.4. Estimation of Modifiers from Past Operating Points

- (i) there exist $\rho_\mu > 0$ such that the inner loop of Algorithm 4.3 converges to the optimum of Problem (4.8) at every RTO iteration k ;
- (ii) the plant cost decreases monotonically at each RTO iteration if $\delta^{\Phi_i} - 3\delta_i^{\Phi_m} - \delta^{\Phi_p} \geq 0, \forall i \in \mathcal{N}$; and
- (iii) there exist a positive constant $\delta_{i,j}^\Delta \in \mathbb{R}$, such that all RTO iterates satisfy the plant constraints (4.1c) if $\delta^{G_{i,j}} > \delta_{i,j}^\Delta, j = 1, \dots, n_{G_i}, \forall i \in \mathcal{N}$.

Proof. The proof can be found in Appendix C.5. □

Remark 4.5 (Convergence of inner-loop algorithm).

It has been shown that adding the second-order terms to the cost and constraints in (4.11) enforces concavity of the dual function for each sub-problem (Necoara et al., 2009). Hence, a dual fast gradient (DFG) algorithm can be implemented for the update of $\boldsymbol{\mu}$ for fast convergence of inner-loop algorithm (Necoara et al., 2011). □

Putting Proposition 4.3 differently, Algorithm 4.3 proposes a feasible-side MA scheme that exploits the knowledge of the interconnection model and local measurements. As already mentioned in Assumption 4.4, the presented algorithm requires utilization of convex nominal models. As can be seen in Algorithm 4.3, the existence of the coordinator is needed to ensure meeting the coupled equality constraints (4.8d) by iteratively adjusting the dual variables. Based on these dual variables, the subsystem optimizers solve the N decoupled optimization problems. This is the inner-loop part of the algorithm, whose iterates are cheap in comparison to the RTO iterates k in the outer loop, the latter involving an experimental setpoint change in the plant.

4.4 Estimation of Modifiers from Past Operating Points

Let us now discuss the approach for obtaining local modifiers and upper bound on the gradient error norm of Lagrangian function for the plant with interconnected systems described in Section 4.1.1.

Here, the method based on linear interpolation or linear regression introduced in Section 2.2.5 is leveraged for interconnected systems. At the k -th RTO iteration, the following input matrix can be constructed for i -th unit

$$\mathbf{U}_{i,k}(\mathbf{z}_{p,i,k}) = [\mathbf{z}_{p,i,k} - \mathbf{z}_{p,i,k-1}, \dots, \mathbf{z}_{p,i,k} - \mathbf{z}_{p,i,k-n_{z_i}}]^\top \in \mathbb{R}^{n_{z_i} \times n_{z_i}}. \quad (4.15a)$$

Assuming that measurements of the cost $\Phi_{p,i}$ and constraints are available at each iteration, we construct the following vectors at the k -th RTO iteration, as well as their model counter-

parts:

$$\begin{aligned}\delta\tilde{\Phi}_{p,i} &= [\tilde{\Phi}_{p,i,k} - \tilde{\Phi}_{p,i,k-1}, \dots, \tilde{\Phi}_{p,i,k} - \tilde{\Phi}_{p,i,k-n_{z_i}}]^\top \in \mathbb{R}^{n_{z_i}} \\ \delta\Phi_i &= [\Phi_{i,k} - \Phi_{i,k-1}, \dots, \Phi_{i,k} - \Phi_{i,k-n_{z_i}}]^\top \in \mathbb{R}^{n_{z_i}}\end{aligned}$$

The measured cost has measurement noise v_k

$$\tilde{\Phi}_{p,i,k} = \Phi_{p,i,k} + v_k.$$

If \mathbf{U}_k is nonsingular, then the set of $n_{z_i} + 1$ RTO points $\{z_{k-j}\}_{j=0}^{n_{z_i}}$ is said to be poised for linear interpolation in $\mathbb{R}^{n_{z_i}}$. The cost modifier at $\mathbf{z}_{p,i,k}$ can then be estimated by FDA as follows:

$$\begin{aligned}\lambda_k^{\Phi_i} &= (\delta\tilde{\Phi}_{p,i} - \delta\Phi_i)^\top \mathbf{U}_{i,k}^{-1} \\ &= \widehat{\nabla_{\mathbf{z}_{p,i}} \Phi_{p,i}}(\mathbf{z}_{p,i,k}) - \nabla_{\mathbf{z}_{p,i}} \Phi_i(\mathbf{z}_{p,i,k})\end{aligned}\quad (4.15b)$$

where $\widehat{\nabla_{\mathbf{z}_{p,i}} \Phi_{p,i}}(\mathbf{z}_{p,i,k}) = \delta\tilde{\Phi}_{p,i}^\top \mathbf{U}_{i,k}^{-1}$ and $\nabla_{\mathbf{z}_{p,i}} \Phi_i(\mathbf{z}_{p,i,k}) = \delta\Phi_i^\top \mathbf{U}_{i,k}^{-1}$. The constraint gradient modifiers in (4.8g) and (4.8i) can be computed in a similar way.

Bounds on Gradient Uncertainty

For the purpose of optimization, the perturbations should be selected so as to obtain accurate Lagrangian gradient estimates (Marchetti et al., 2010). An upper bound on the gradient error norm is discussed in Section 2.2.4. Assuming that interconnection constraint in (4.1d) has structure $\mathbf{Q}_p := \mathbf{v}_p - \mathbf{H}\mathbf{y}_p(\mathbf{z}_p) = \mathbf{0}$, the measured value of Lagrangian function for the plant with interconnected systems described in Section 4.1.1, can be expressed as

$$\begin{aligned}\tilde{L}_p(\mathbf{z}_p) &= \sum_{i \in \mathcal{N}} \tilde{\Phi}_{p,i}(\mathbf{z}_{p,i}) + \sum_{i \in \mathcal{N}} \boldsymbol{\mu}_{G_i}^\top \tilde{\mathbf{G}}_{p,i}(\mathbf{z}_{p,i}) + \boldsymbol{\mu}_Q^\top \tilde{\mathbf{Q}}_p(\mathbf{z}_p) \\ &= \sum_{i \in \mathcal{N}} (L_{p,i}(\mathbf{z}_{p,i}) + v_i).\end{aligned}\quad (4.16a)$$

Here v_i denotes the resulting noise in the Lagrangian function of i -th subsystem and $\boldsymbol{\mu}_G, \boldsymbol{\mu}_Q$ being the Lagrange multipliers. The superscript $(\tilde{\cdot})$ denotes noisy measurements. In (4.16a), $L_{p,i}(\mathbf{z}_{p,i}), \forall i \in \mathcal{N}$ are

$$L_{p,i}(\mathbf{z}_{p,i}) = \Phi_{p,i}(\mathbf{z}_{p,i}) + \boldsymbol{\mu}_{G_i}^\top \mathbf{G}_{p,i}(\mathbf{z}_{p,i}) + \boldsymbol{\mu}_{Q,i}^\top \mathbf{v}_{p,i} - \sum_{j \in \mathcal{N}} \boldsymbol{\mu}_{Q,j}^\top \mathbf{H}_{ij} \mathbf{y}_{p,i}(\mathbf{z}_i).$$

Therefore, the i -th gradient $\nabla_{\mathbf{z}_{p,i}} \mathbf{L}_{p,i} \in \mathbb{R}^{1 \times n_{z_i}}, i \in \mathcal{N}$ is given as

$$\nabla_{\mathbf{z}_{p,i}} \mathbf{L}_{p,i} = \nabla_{\mathbf{z}_{p,i}} \Phi_{p,i}(\mathbf{z}_{p,i}) + \boldsymbol{\mu}_{G_i}^\top \nabla_{\mathbf{z}_{p,i}} \mathbf{G}_{p,i}(\mathbf{z}_{p,i}) + \boldsymbol{\mu}_{Q,i}^\top - \sum_{j \in \mathcal{N}} \boldsymbol{\mu}_{Q,j}^\top \mathbf{H}_{ij} \nabla_{\mathbf{z}_i} \mathbf{y}_{p,i}(\mathbf{z}_i).$$

Finally, the gradient estimation error $\mathbf{e}^{L_i}(\mathbf{z}_{p,i}) \in \mathbb{R}^{n_{z_i} \times 1}$ can be stated as

$$\mathbf{e}^{L_{p,i}}(\mathbf{z}_{p,i}) = \mathbf{e}^{\Phi_{p,i}}(\mathbf{z}_{p,i}) + \boldsymbol{\mu}_{G_i}^\top \mathbf{e}^{G_{p,i}}(\mathbf{z}_{p,i}) - \sum_{j \in \mathcal{N}} \boldsymbol{\mu}_{Q,j}^\top \mathbf{H}_{ij} \mathbf{e}^{Y_{p,i}}(\mathbf{z}_{p,i}) \quad (4.18)$$

where $\mathbf{e}^{\Phi_{p,i}}(\mathbf{z}_{p,i})$, $\mathbf{e}^{G_{p,i}}(\mathbf{z}_{p,i})$ and $\mathbf{e}^{Y_{p,i}}(\mathbf{z}_{p,i})$ are gradient estimation error of the cost $\Phi_{p,i}$, constraint $\mathbf{G}_{p,i}$ and outputs $\mathbf{y}_{p,i}$, respectively. As already explained in Section 2.2.4, each error function is defined as difference between the estimated gradient and the true plant gradient which can be split in truncation and measurement noise error as in (2.14b).

Let $L_{p,i}(\mathbf{z}_{p,i})$ be a twice continuously differentiable function, then the norm of the gradient error $\mathbf{e}^{L_{p,i}}(\mathbf{z}_{p,i})$ in (4.18) can be upper bounded as follows

$$\begin{aligned} \|\mathbf{e}^{L_{p,i}}\| &\leq \|\mathbf{e}^{\Phi_{p,i}}\| + \|\boldsymbol{\mu}_{G_i}\| \|\mathbf{e}^{G_{p,i}}\| + \sum_{j \in \mathcal{N}} \|\boldsymbol{\mu}_{Q,i}\| \|\mathbf{H}_{ij}\| \|\mathbf{e}^{Y_{p,i}}\| \\ &\leq \mathcal{E}^{L_{p,i}} := \mathcal{E}^{\Phi_{p,i}} + \|\boldsymbol{\mu}_{G_i}\| \mathcal{E}^{G_{p,i}} + \sum_{j \in \mathcal{N}} \|\boldsymbol{\mu}_{Q,i}\| \|\mathbf{H}_{ij}\| \mathcal{E}^{Y_{p,i}} \end{aligned} \quad (4.19)$$

where $\mathcal{E}^{\Phi_{p,i}}$, $\mathcal{E}^{G_{p,i}}$ and $\mathcal{E}^{Y_{p,i}}$ are the bounds on gradient error of plant functions $\Phi_{p,i}$, $\mathbf{G}_{p,i}$ and $\mathbf{y}_{p,i}$. These bounds are obtained as in (2.15) in Subsection 2.2.4.

4.5 Examples

Next, we present two examples. Subsections 4.5.1 and 4.5.2 discuss the application of Algorithms 4.2 and 4.3, respectively. Here, we consider that the plant gradient estimates are exact.

4.5.1 Numerical Example

We illustrate the essential features of the proposed distributed modifier-adaptation schemes via a numerical example. We study a system consisting of two interconnected subsystems as depicted in Figure 4.3. The interconnection structure is

$$v_1 - y_{21} = 0, \quad (4.20a)$$

$$v_2 - y_{11} = 0. \quad (4.20b)$$

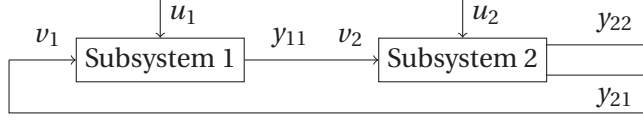


Figure 4.3 – Input-output structure of two interconnected subsystems.

The objective is to minimize the total cost $\sum_{i=1}^2 \Phi_i$ of the plant defined next:

$$y_{p,11} = 2.6u_1 + 0.1v_{p,1}^2, \quad (4.21a)$$

$$y_{p,21} = 0.2e^{u_2} - 0.1u_2v_{p,2}, \quad (4.21b)$$

$$y_{p,22} = 0.1u_2v_{p,2} - 0.2u_2^2, \quad (4.21c)$$

$$G_{p,1} = u_1^2 - v_{p,2} - 0.5 \leq 0, \quad (4.21d)$$

$$\Phi_{p,1} = -6u_1^2 - 5u_1v_{p,1} + (y_{p,11} - 1)^2, \quad (4.21e)$$

$$\Phi_{p,2} = u_2^2 + 4u_2 + 0.5(y_{p,22})^2. \quad (4.21f)$$

The available model reads:

$$y_{11} = 2.2u_1 + 0.6v_1, \quad (4.22a)$$

$$y_{21} = 0.63u_2 + 0.5v_2, \quad (4.22b)$$

$$y_{22} = 2.4u_2 + 2.2v_2, \quad (4.22c)$$

$$G_1 = u_1 - v_2 \leq 0, \quad (4.22d)$$

$$\Phi_1 = u_1^2 + 6u_1 + (y_{11} - 1)^2, \quad (4.22e)$$

$$\Phi_2 = u_2^2 + 4u_2 + 0.5(y_{22})^2. \quad (4.22f)$$

The plant optimum is $\mathbf{u}_p^* = (1.0998, -2.3744)$, while the optimal input for the nominal model is $\mathbf{u}^* = (0.1190, -0.1071)$. We assume that exact plant gradients and noise-free measurements are available.

The optimum solution to the plant problem (4.20) and (4.21) is obtained by using the model equations (4.20), (4.22) and distributed MA Algorithm 4.2. As the model optimization problem satisfies Assumption 4.4, dual decomposition with $\rho_\mu = 1$ is used. Moreover, we use the filter gain $\mathbf{K} = 0.8\mathbf{I}$.

The resulting sequences of inputs and interconnection variables are shown in Figure 4.4. The algorithm converges to the optimal plant values for both \mathbf{u} and \mathbf{v} . For every RTO iteration, the number of model-based inner iterations is shown in Figure 4.5. The algorithm was also tested in the presence of errors on the plant cost and constraint gradients. Additive zero-mean errors with 8% standard deviation were considered. In this case, the input sequence converges to a small neighborhood of the plant optimum, which is similar to the observation made by Marchetti et al. (2009).

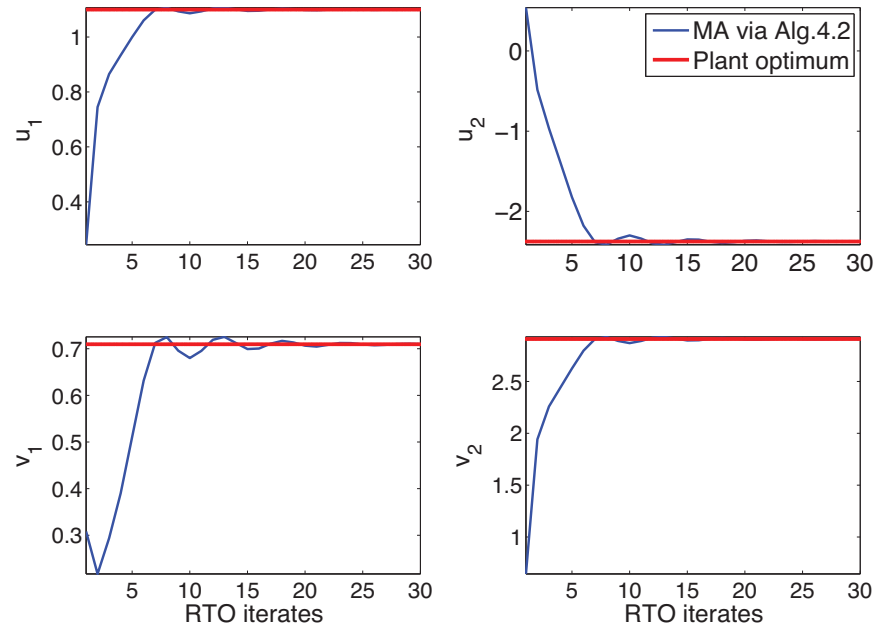


Figure 4.4 – Convergence of the input variables \mathbf{u} and interconnection variables \mathbf{v} via Algorithm 4.2.

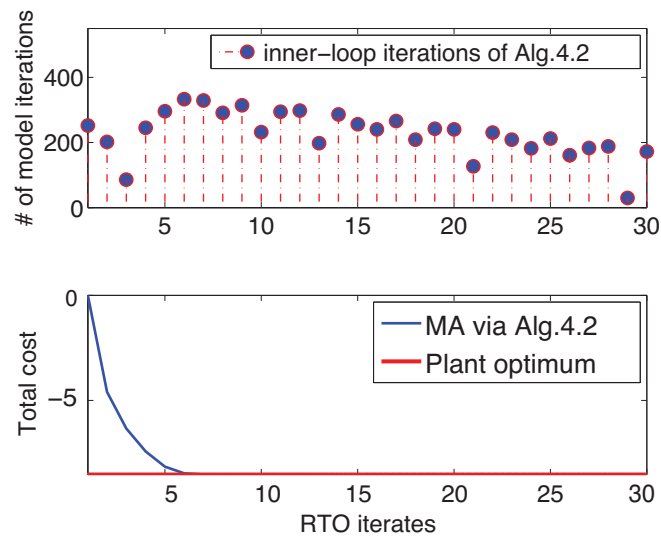


Figure 4.5 – Number of model-based inner-loop iterations of Algorithm 4.2 (top) and plant cost evolution (bottom).

4.5.2 Optimization of Serial Gas-Compressor Station

In the following example the problem of optimal load sharing of serial gas compressors in the presence of plant-model mismatch has been tackled. We analyze the station with two compressors in series due to the simplicity but it can be easily scaled to the case of N compressors. We tackle this problem via Algorithm 4.3. The main advantage of the algorithm is that it guarantees convergence to the plant optimum by requiring only local estimation of the modifiers.

Consider a station with two compressors in series as shown in Figure 4.6. The optimal setpoints for the speed ω_i and the recycle valve opening $V_{rec,i}$, with $i \in \mathcal{N} = \{1,2\}$, can be obtained by minimizing the sum of the power consumption of the two compressors. It is assumed that the station is operated by keeping the inlet $V_{in,i}$ and outlet $V_{out,i}$ valves at constant values. In the serial configuration, the two compressors are arranged such that the discharge tank of the first compressor feeds into the suction tank of the second compressor, as shown in Figure 4.6. Hence, the interconnection relationship of the compressors reads $p_{in,2} = p_{d,1}$. It is necessary that the pressure downstream of Compressor 2 be above a specified pressure setpoint $p_{d,2}^{sp}$. Mathematically, the centralized RTO problem can be stated as follows:

$$\min_{\omega_1, \omega_2} \Phi_1 + \Phi_2 \quad (4.23a)$$

s.t. steady-state model equations

$$s_{0,i} - s_{1,i} m_{c,i} + \Pi_i \leq 0, \quad i \in \{1,2\} \quad (4.23b)$$

$$c_{0,i} + c_{1,i} m_{c,i} - \Pi_i \leq 0, \quad i \in \{1,2\} \quad (4.23c)$$

$$0 \leq V_{rec,i} \leq V_{rec,i}^U, \quad i \in \{1,2\} \quad (4.23d)$$

$$\omega_i^L \leq \omega_i \leq \omega_i^U, \quad i \in \{1,2\} \quad (4.23e)$$

$$p_{d,2}^{sp} - p_{d,2} \leq 0 \quad (4.23f)$$

$$p_{in,2} = p_{d,1}, \quad (4.23g)$$

where $s_{0,i}$ and $s_{1,i}$ are positive constants that define surge constraints (4.23b). The violation of this constraint is prevented with anti-surge controller (ASC). Furthermore, $c_{0,i}$ and $c_{1,i}$ define choke constraints (4.23c). Also, there are lower and upper bounds for the recycle valve openings $V_{rec,i}$ in (4.23d), as well as for the speed ω_i in (4.23e).

Load-sharing optimization via distributed MA Algorithm 4.3

For simplicity, the analysis is restricted to station flow setpoints $p_{d,2}^{sp}$ for which all compressors operate far from the surge line, that is, without the need for anti-surge control. In other words, we assume $V_{rec,1} = V_{rec,2} = 0$. The interconnection variable $v_2 = p_{in,2}$ is given by the interconnection relationship (4.23g), where the output of Compressor 1 is $y_1 = p_{d,1}$. Hence, the interconnection model (4.8d) for the system presented in Figure 4.6 is given as $v_2 = Hy_1$,

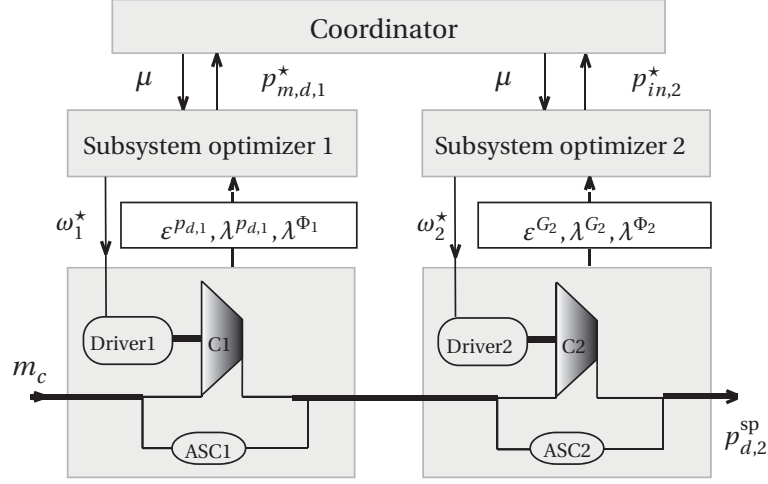


Figure 4.6 – Schematic diagram of serial compressors.

where $H = 1$. Also, the local inputs for each subsystem are defined as

$$\mathbf{z}_1^\top = (\omega_1, p_{in,1}), \quad \mathbf{z}_2^\top = (\omega_2, p_{in,2}). \quad (4.24)$$

Plant-model mismatch is introduced by using different compressor maps (2.24) for the model and the plant. The efficiency maps (A and B) for two different plants, shown in Figure 4.7, are used in (2.25a) to obtain the plant cost functions $\Phi_{p,1}$ and $\Phi_{p,2}$. The model cost functions Φ_1 and Φ_2 are obtained using the model efficiency map in Figure 4.7. Enforcing plant optimality via Algorithm 4.3 requires the use of convex cost models and linear constraints with respect to the local inputs. Hence, linearized model equations and constraints of Problem (4.23) are used as well as the convex cost models Φ_i^c of Φ_i with respect to \mathbf{z}_i at the initial operating points $\mathbf{z}_{p,1,0}$ and $\mathbf{z}_{p,2,0}$ of Compressor 1 and Compressor 2, respectively.

For each compressor, the local optimization problem is used in the inner-loop of Algorithm 4.3:

- (i) The optimization problem for the first compressor is defined according to (4.13):

$$\mathbf{z}_{1,k+1}^{l,*} = \underset{\mathbf{z}_1 \in \mathcal{F}_{1,k}^U}{\operatorname{argmin}} \Phi_1^c(\mathbf{z}_1) + \left(\boldsymbol{\lambda}_k^{\Phi_1} \right)^\top (\mathbf{z}_1 - \mathbf{z}_{p,1,k}) + \frac{\delta^{\Phi_1}}{2} \|\mathbf{z}_1 - \mathbf{z}_{p,1,k}\|_2^2 - \mu^l p_{d,m,1}(\mathbf{z}_1), \quad (4.25a)$$

where μ are the Lagrange multipliers of the interconnection constraint (4.23g) and \mathcal{F}_1^U is the feasible set of Compressor 1 defined via box constraints on the speed ω_1 in (4.23e).

Also, $p_{d,m,1}(\mathbf{z}_1)$ is the modified discharge pressure function:

$$p_{d,m,1}(\mathbf{z}_1) = p_{d,1}(\mathbf{z}_1) + \epsilon_k^{p_{d,1}} + \boldsymbol{\lambda}_k^{p_{d,1}} (\mathbf{z}_1 - \mathbf{z}_{p,1,k})$$

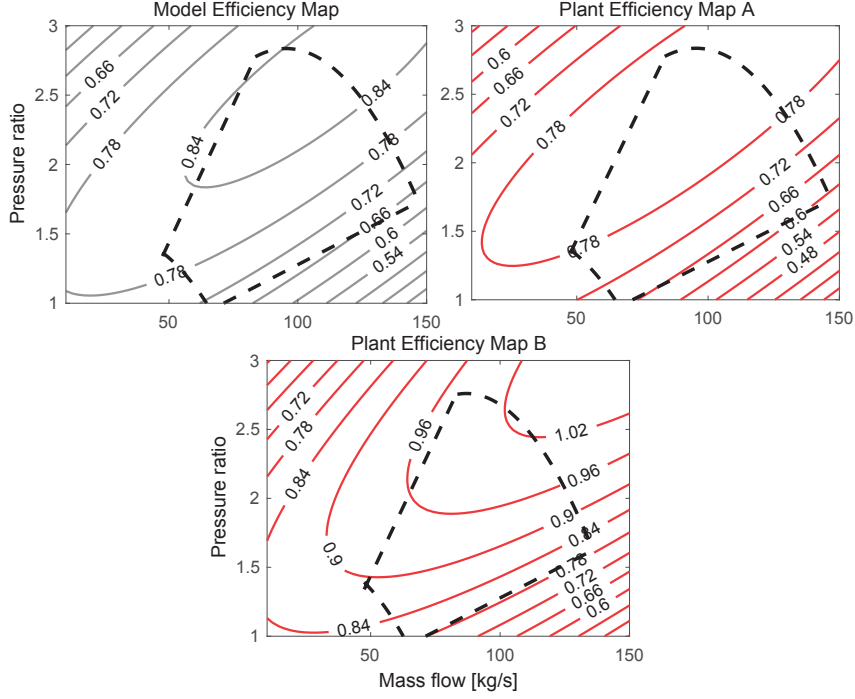


Figure 4.7 – Comparison between the model efficiency map (grey) and the plant efficiency maps (red) for two different compressors. The maps are shown as contour plots, whereas the dashed black lines represent the operating range of the gas compressor.

and modifiers for the first compressors read:

$$\begin{aligned}\varepsilon_k^{p_{d,1}} &= p_{d,p}(\mathbf{z}_{p,1,k}) - p_d(\mathbf{z}_{p,1,k}) \\ \boldsymbol{\lambda}_k^{\Phi_1} &= \nabla_{\mathbf{z}_1} \Phi_{p,1}(\mathbf{z}_{p,1,k}) - \nabla_{\mathbf{z}_1} \Phi_1^c(\mathbf{z}_{p,1,k}) \\ \boldsymbol{\lambda}_k^{p_{d,1}} &= \nabla_{\mathbf{z}_1} p_{d,p,1}(\mathbf{z}_{p,1,k}) - \nabla_{\mathbf{z}_1} p_{d,1}(\mathbf{z}_{p,1,k}).\end{aligned}$$

Here, $\nabla_{\mathbf{z}_1} \Phi_{p,1}(\mathbf{z}_{p,1,k})$ and $\nabla_{\mathbf{z}_1} p_{d,p,1}(\mathbf{z}_{p,1,k})$ are the derivatives with respect to \mathbf{z}_1 of the plant cost $\Phi_{p,1}$ and plant outlet pressure $p_{d,p,1}$, respectively, evaluated at $\mathbf{z}_{p,1,k}$. $\boldsymbol{\lambda}^{\Phi_1}$ and $\boldsymbol{\lambda}^{p_{d,1}}$ are the cost and outlet pressure modifiers, respectively.

(ii) The optimization problem for the second compressor reads:

$$\mathbf{z}_{2,k+1}^{l,*} = \underset{\mathbf{z}_2 \in \mathcal{F}_{2,k}^U}{\operatorname{argmin}} \Phi_2^c(\mathbf{z}_2) + \left(\boldsymbol{\lambda}_k^{\Phi_2}\right)^\top (\mathbf{z}_2 - \mathbf{z}_{p,2,k}) + \frac{\delta^{\Phi_2}}{2} \|\mathbf{z}_2 - \mathbf{z}_{p,2,k}\|_2^2 + \mu^l p_{in,2}(\mathbf{z}_2), \quad (4.26a)$$

with $\mathcal{F}_{2,k}^U$ being the feasible set of Compressor 2 defined as $\mathcal{F}_{2,k}^U = \mathcal{U}_{\omega_2} \cap G_{2,k}^U$, where

\mathcal{U}_{ω_2} denote box constraints on the speed ω_2 (4.23e) and $G_{2,k}^U$ is defined as

$$G_{2,k}^U := G_2(\mathbf{z}_2) + \boldsymbol{\varepsilon}_k^{G_2} + \left(\boldsymbol{\lambda}_k^{G_2}\right)^\top (\mathbf{z}_2 - \mathbf{z}_{p,2,k}) + \frac{\delta^{G_2}}{2} \|\mathbf{z}_2 - \mathbf{z}_{p,2,k}\|_2^2 \leq 0 \quad (4.26b)$$

$$G_2(\mathbf{z}_2) := p_{d,2}^{\text{sp}} - p_{d,2} \leq 0 \quad (4.26c)$$

$$\boldsymbol{\lambda}_k^{G_2} = \nabla_{\mathbf{z}_2} G_{p,2,k}(\mathbf{z}_{p,2,k}) - \nabla_{\mathbf{z}_2} G_{2,k}(\mathbf{z}_{p,2,k}),$$

where $\boldsymbol{\varepsilon}_k^{G_2}$ and $\boldsymbol{\lambda}_k^{G_2}$ are the zero-order and first-order modifiers of the plant constraint $G_{p,2}$, respectively, evaluated at $\mathbf{z}_{p,2,k}$. Also, the cost first-order modifier is given as:

$$\boldsymbol{\lambda}_k^{\Phi_2} = \nabla_{\mathbf{z}_2} \Phi_{p,2}(\mathbf{z}_{p,2,k}) - \nabla_{\mathbf{z}_2} \Phi_2^c(\mathbf{z}_{p,2,k}).$$

(iii) We define next the (b) and (c) steps of the inner loop of Algorithm 4.3:

The solutions $\mathbf{z}_{1,k+1}^{l,*}$ and $\mathbf{z}_{2,k+1}^{l,*}$ from (4.25a) and (4.26a) are used to update the dual variable μ as follows:

$$\mu^{l+1} = \mu^l + \rho_\mu \left(p_{in,2,k+1}^{l,*} - p_{d,m,1,k+1}^{l,*} \right),$$

where

$$p_{d,m,1,k+1}^{l,*} = p_{d,1}(\mathbf{z}_{p,1,k+1}^{l,*}) + \varepsilon^{p_{d,1}} + \left(\boldsymbol{\lambda}_k^{p_{d,1}}\right)^\top \left(\mathbf{z}_{p,1,k+1}^{l,*} - \mathbf{z}_{p,1,k}\right).$$

Simulation results

The distributed MA Algorithm 4.3 is tested in simulation. The scenario involves steady-state optimization of a station composed of two serial compressors, as presented in Figure 4.6.

We assume that gradient information regarding the plant costs $\Phi_{p,1}$ and $\Phi_{p,2}$, the constraint G_2 and the output $p_{d,1}$ is available. The convergence of the plant inputs to the optimum is presented in Figure 4.8. Figure 4.9 shows the performance loss

$$\Delta P_k = \frac{\sum_{i=1}^N \Phi_{p,i}(\mathbf{z}_{p,i,k}) - \sum_{i=1}^N \Phi_{p,i}(\mathbf{z}_{p,i}^*)}{\sum_{i=1}^N \Phi_{p,i}(\mathbf{z}_{p,i}^*)} \times 100\%,$$

where $\sum_{i=1}^N \Phi_{p,i}(\mathbf{z}_{p,i}^*)$ denotes the true optimal cost of the plant, and $\sum_{i=1}^N \Phi_{p,i}(\mathbf{z}_{p,i,k})$ is the plant cost obtained at the k -th RTO iterate. We also compare the performance of distributed MA with that of equal-load distribution, which corresponds to current industrial practice. Note that the benefit of applying the RTO algorithm is around 6.5% and 2.4% in comparison to the initial point and equal-load distribution, respectively. Furthermore, Figure 4.9 depicts the number of inner-loop iterations, which are purely model-based and thus cheap compared to the outer RTO iterations that involve experimental steady-state transitions. We observe monotonic decrease of the plant cost and feasibility through the RTO iterations, as expected from the properties of the distributed MA algorithm. The initial point is chosen

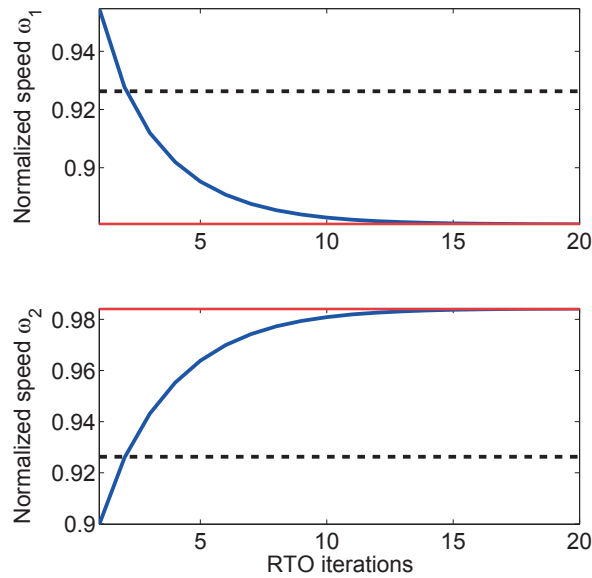


Figure 4.8 – (Blue) Evolution of the inputs (speeds) via Algorithm 4.3. (Red) Plant optimum. (Dashed black) Speed for equal-load distribution.

such that the plant constraint $G_2(\mathbf{z}_2)$ is satisfied. In particular, Figure 4.10 shows that the outlet pressure $p_{d,2}$ of the downstream compressor exceeds $p_{d,2}^{sp}$ through all RTO iterations. Also, constraint $G_2(\mathbf{z}_2)$ is active at the plant optimum.

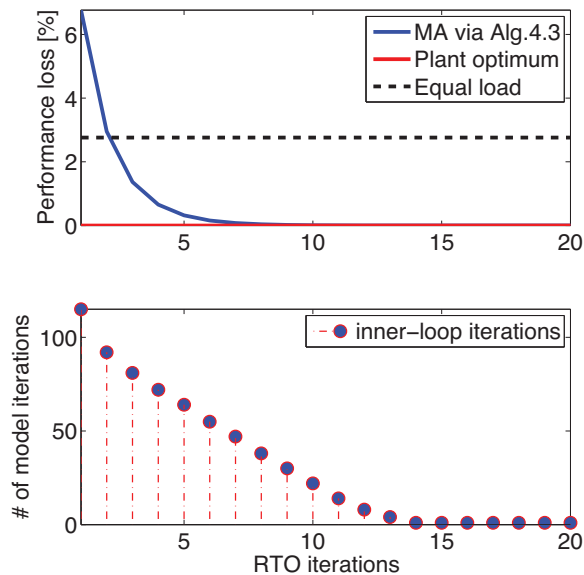


Figure 4.9 – The compressors station performance loss via Algorithm 4.3 (top) and number of model-based inner-loop iterations (bottom).

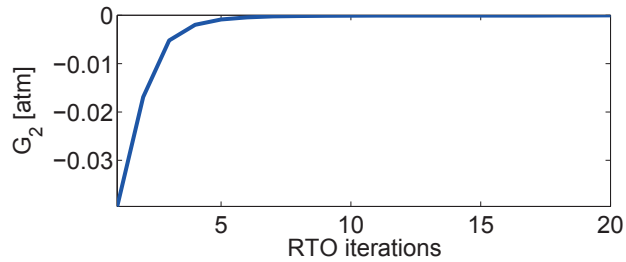


Figure 4.10 – Value of the constraint $G_2(\mathbf{z}_2)$ through RTO iterations of Algorithm 4.3.

4.6 Conclusion

This chapter has discussed the convergence properties of centralized and distributed modifier adaptation algorithm for the real-time optimization of uncertain interconnected systems. The proposed modifier-adaptation schemes use local plant gradient information with respect to both the decision variables *and* the interconnection variables. Because of this property, this algorithm overcomes the challenges of most distributed algorithms in the context of RTO, in particular the dependence on global plant gradients and on the availability of a global model. This is possible because the algorithms use the measurement of interconnection variables and can handle uncertainty for each subsystem locally, that is, without the need to exchange plant gradient information among subsystems. Furthermore, feasible-side convergence of a distributed modifier scheme was shown, provided perfect plant gradient information is available. The effectiveness of the algorithm was illustrated on load-sharing optimization problem for serial compressors.

From an RTO point of view, the main benefit of both formulations is that the plant sensitivities with respect to only local inputs of each subsystem need to be estimated. Potentially, this can reduce the required number of perturbations for gradient estimation. However, the gradient estimates with respect to the interconnection variables are required. The problem is that these variables are not directly manipulated, and careful steps have to be taken in the perturbation design procedure for interconnected systems.

5 Application to Gas-Compressor Stations

This chapter is based on:

P. Milosavljevic, A. Cortinovis, A. Marchetti, T. Faulwasser, M. Mercangöz, and D. Bonvin. Optimal load sharing of parallel compressors via modifier adaptation. In *IEEE Conference on Control Applications*, pages 1488–1493, 2016.

P. Milosavljevic, A. Cortinovis, R. Schneider, T. Faulwasser, M. Mercangöz, and D. Bonvin. Optimal load sharing for serial compressors via modifier adaptation. In *European Control Conference (ECC)*, pages 2306–2311, 2018b.

P. Milosavljevic, A. Cortinovis, A. Marchetti, T. Faulwasser, M. Mercangöz, and D. Bonvin. Load-sharing optimization of parallel and serial compressors via modifier adaptation. *Applied Energy*, (in preparation), 2018.

Compressors in industrial applications are most commonly operated in stations comprising two main arrangements, that is, parallel and serial configurations. While parallel centrifugal compressors are used in applications that demand high mass flowrates and low pressure ratios (Boyce, 2003), serial compressor configurations are employed to increase the total discharge pressure compared to the pressure that a single compressor can deliver. Typically, a compressor station receives a reference signal from the dispatch center, usually in the form of a total mass-flow setpoint for parallel configurations, or a pressure setpoint for serial configuration. Often, these setpoints are tracked by applying an equal load distribution to each compressor. In turn, the individual compressor control systems track their local setpoints by adjusting the operating speeds. Usually, compressor stations consist of heterogeneous units. Yet, the industrial practice of equal load distribution does not account for differences in the efficiencies. Hence, it often results in suboptimal operation.

The issue of suboptimal operation can be approached by solving an optimization problem in order to select the number of active compressors (Xenos et al., 2015) and to assign

mass flows to the compressors, that is, allocating speeds to the drives (Cortinovic et al., 2016; Kumar and Cortinovic, 2017). This problem is often referred to as the load-sharing optimization problem (Nguyen et al., 2008; Kumar and Cortinovic, 2017; Xenos et al., 2015). The main challenge of load-sharing optimization is that the efficiency maps of the compressors are subject to considerable uncertainty. Moreover, these maps change over time due to erosion and fouling (Kurz and Brun, 2012a). To that end, several groups have recently investigated different strategies for load-sharing optimization. These range from methods to update efficiency maps over time via surrogate models and process data (Tirnovan et al., 2008), through data-driven models (Xenos et al., 2015) to an online optimization framework relying on recursive updates of model parameters (Paparella et al., 2013). In current industrial practice, the uncertainty surrounding the efficiency of compressor maps is mainly addressed using the so-called two-step approach (Cortinovic et al., 2016; Kumar and Cortinovic, 2017; Xenos et al., 2015), that is, the update of compressor maps via static nonlinear regression followed by load-sharing optimization.

In this chapter, the problem of distributing a load among the available machines in a parallel and serial compressor station is considered when the compressor efficiency characteristics are not precisely known. This problem is tackled via the centralized first-order MA Algorithm 4.1 for interconnected systems that uses local input and output measurements of each subsystem. The most challenging part of MA lies in the fact that the gradients of the plant outputs with respect to the inputs must be estimated. Here, the gradient estimation of compressors' cost and constraints with respect to local inputs exploiting the interconnection structure is discussed. In addition, the gradient estimation is elaborated when the plant is operated close to the surge conditions. Such conditions are particularly challenging due to the discontinuity in gradient that occurs. In this thesis, the following characteristics are explicitly considered: (i) fixed number of active units, (ii) parallel and serial connections of gas compressors, (iii) operation close to the surge constraints, and (iv) availability of *uncertain* individual compressor performance maps. Moreover, the analysis shows how the interconnection structure of the compressor station can be exploited to overcome the hurdle of estimating process gradients. Furthermore, the complexity of this estimation is independent of the number of compressors. The real-time optimization algorithm works in synergy with a low-level controller that tracks the required discharge pressure in serial configuration or the required mass flow in parallel configuration.

This chapter is organized as follows. Section 5.1 formulates the load-sharing optimization problem for a parallel and serial-compressor plant. Section 5.2 proposes an MA solution to the optimal load-sharing problem. The gradient estimation technique applied to functions with discontinuities in the gradient is presented in Section 5.3. A case study is presented in Section 5.4, and Section 5.5 concludes this chapter.

5.1 Problem Formulation

Here, the compressor model described in Section 2.3 is considered. Namely, the dynamics of the compressor is presented in (2.22) while the compressor maps and the cost function are given in (2.24) and (2.25a), respectively.

Subsection 5.1.1 gives details on low level controllers used in both configurations. Furthermore, the load-sharing optimization (LSO) problem for both, parallel and serial configuration of compressors operating in synergy with process controller is presented in Subsection 5.1.2 and Subsection 5.1.3, respectively.

5.1.1 Underlying Control Loops

Before detailing the specifics of parallel and serial compressor configurations, a few comments on the control loops, depicted in Figure 5.1, are in order. Consider a station with N separate gas compressors connected in either a parallel or a serial arrangement, as depicted in Figure 5.1. Subsequently, we assumed that all compressors are active and operate continuously for the time period under investigation.

Each compressor uses a local anti-surge controller to avoid surge conditions. The surge control line distance is controlled by means of a PI controller manipulating the opening of the anti-surge control valve. The control signal is constrained between 0 and 100% (valve opening) and is only giving a control action different from zero if the surge control line distance becomes positive. In addition, another PI controller computes the speed setpoint for each driver in order to track the station demand provided by the user, either $m_{tot}^{sp}(t)$ or $p_{d,N}^{sp}(t)$.

In absence of an optimization layer, all parallel compressors would receive the same speed setpoints, thus implementing suboptimal load distribution among the interconnected compressors since the compressors might be of different sizes, different ages, different states of maintenance.

5.1.2 Parallel Configuration

Consider a station with N parallel compressors, as illustrated in Figure 5.1(top), and let $i \in \mathcal{N} = \{1, \dots, N\}$ be the corresponding index set. The optimal setpoints for the inputs ω_i can be obtained by minimizing the total power consumption of all compressors. In the parallel arrangement, the inlet and outlet pressures, $p_{in,i}$ and $p_{out,i}$, are the same for all compressors and considered to be constant. In contrast, the mass flows $m_{c,i}$, $\forall i \in \mathcal{N}$, can be varied independently by manipulating the speeds ω_i . This is possible due to variable-speed drives. The individual output flows $m_{out,i}$ of all compressors must add up to the station flow setpoint m_{tot}^{sp} in order to meet the plant demand. Under normal process operation, the power consumption is minimized by keeping the inlet and outlet valves completely open, that is,

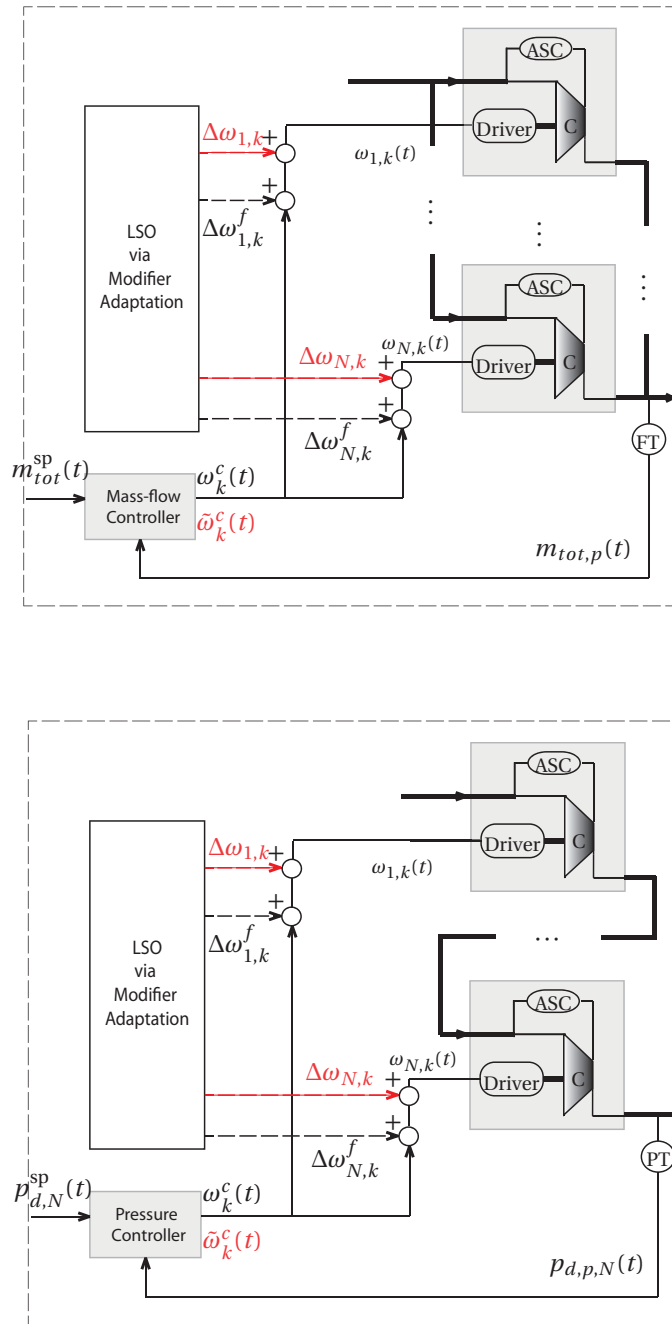


Figure 5.1 – Diagram of a parallel-compressor configuration with mass-flow control (**top plot**); and serial-compressor configuration with pressure control (**bottom plot**). PT and FT represent pressure and flow transmitters, respectively. ASC is an anti-surge controller. The control is executed in continuous time t , while the optimization problem is executed only once the system has reached steady state. $\Delta\omega_{i,k}^f$ is the contribution of the optimization layer to the rotational speed $\omega_{i,k}$, while $\Delta\omega_{i,k}$ is the perturbation that is added to compute the experimental gradients. ω_k^c is the speed reached at the k -th RTO iteration without the perturbation $\Delta\omega_{i,k}$, while $\tilde{\omega}_k^c$ is the speed reached at steady state with the perturbations $\Delta\omega_{i,k}$ added. LSO: load-sharing optimization

$V_{in,i} = V_{out,i} = 1$. Hence, the problem can be stated as follows

$$\min_{\omega, V_{rec}} \sum_{i \in \mathcal{N}} \Phi_i \quad (5.1a)$$

s.t. steady-state equations obtained from (2.22)-(2.24)

$$m_{tot} = \sum_{i \in \mathcal{N}} m_{out,i} = m_{tot}^{sp} \quad (5.1b)$$

$$G_{1,i} = \frac{1}{s_{1,i}} (s_{0,i} + \Pi_i) - m_{c,i} - s_{2,i} \leq 0, \quad (5.1c)$$

$$G_{2,i} = \frac{1}{c_{1,i}} (c_{0,i} - \Pi_i) + m_{c,i} - c_{2,i} \leq 0, \quad (5.1d)$$

$$m_{c,i} \in \mathcal{M}_i, \omega_i \in \mathcal{W}_i, V_{rec,i} \in \mathcal{V}_{rec,i}, i \in \mathcal{N}, \quad (5.1e)$$

where m_{tot} is the total output flow, and $s_{0,i}$, $s_{1,i}$ and $s_{2,i}$ are the surge line coefficients of the i -th compressor. The surge constraint $G_{1,i}$ expresses a lower limit on the mass flow of the i -th compressor for a given head or pressure ratio. If this limit is violated, then a flow instability called surge occurs, which can cause thermal and mechanical stress to compressor blades, potentially leading to damages and eventually also to machine failure. Each compressor uses a local anti-surge controller in order to avoid surge conditions by manipulating the recycle valve openings $V_{rec,i}$ within bounds defined in (5.1e). The choke line coefficients of the i -th compressor are $c_{0,i}$, $c_{1,i}$ and $c_{2,i}$. The choke constraint $G_{2,i}$ corresponds to the maximum flow that can be generated by the i th compressor depending on the aerodynamic characteristics of the compressor and the discharge piping. Box constraints on the operating flows, speeds and valve openings are included in (5.1e).

5.1.3 Serial Configuration

Consider a station with N compressors operating in a serial configuration as illustrated in Figure 5.1(bottom). In the serial setup, the compressors are arranged such that the discharge of the i -th compressor feeds into the suction of the $(i + 1)$ -st compressor. The first compressor is connected to the suction header, while the last compressor is connected to the discharge header. Hence, only the inlet boundary pressure of the first compressor $p_{in,1}$ and the outlet boundary pressure of the last compressor $p_{out,N}$ are considered to be specified. In contrast to the parallel topology, the flow through each compressor $m_{out,i}$, $\forall i \in \mathcal{N}$, is the same. Since variable-speed drives are considered, the individual pressure ratios Π_i can be varied independently by manipulating the speeds ω_i . The optimal speed values can be computed by minimizing the total power consumption of all compressors. Here as well, we consider that the power consumption is minimized by keeping the inlet and outlet valves completely open. The discharge and suction lines of two neighboring compressors share the same valve, and their interconnection can be expressed as $p_{d,i} = p_{in,i+1}$ with $m_{out,i} = m_{in,i+1}$. It is desired to enforce that the discharge pressure of the N -th compressor be equal to the pressure setpoint

$p_{d,N}^{\text{SP}}$. A suitable load-sharing optimization problem reads as follows

$$\min_{\omega, V_{rec}} \sum_{i \in \mathcal{N}} \Phi_i \quad (5.2a)$$

s.t. steady-state equations obtained from (2.22)-(2.24)

$$p_{d,N} = p_{d,N}^{\text{SP}} \quad (5.2b)$$

$$p_{d,i} = p_{in,i+1} \quad i \in \{1, \dots, N-1\} \quad (5.2c)$$

$$m_{out,i} = m_{in,i+1} \quad i \in \{1, \dots, N-1\}, \quad (5.2d)$$

$$G_{1,i} = \frac{1}{s_{1,i}} (s_{0,i} + \Pi_i) - m_{c,i} - s_{2,i} \leq 0, \quad (5.2e)$$

$$G_{2,i} = \frac{1}{c_{1,i}} (c_{0,i} - \Pi_i) + m_{c,i} - c_{2,i} \leq 0, \quad (5.2f)$$

$$m_{c,i} \in \mathcal{M}_i, \omega_i \in \mathcal{W}_i, V_{rec,i} \in \mathcal{V}_{rec,i}, i \in \mathcal{N}, \quad (5.2g)$$

where the constraints (5.2e)–(5.2g) are the same as the constraints (5.1c)–(5.1e) in the load-sharing optimization problem for the parallel configuration.

5.2 Load-Sharing Optimization via MA

Compressors in a compressor plant may operate at different points on compressor map. This also includes operating points close to surge conditions. This requires modeling the discontinuity that occurs when the anti-surge controllers become active or inactive. In this study a fixed number of active compressors is assumed.

We assume that the choke constraints in (5.1d) and (5.2f) do not become active in the analysis that follows. The reason is because the most efficient operating conditions for both configurations typically involve the activation of the surge constraints, while operating on the choke line is not economically efficient. Note, however, that the formulation of load-sharing optimization for the choke conditions can be handled in a similar way.

5.2.1 MA for Parallel Compressors

As explained in Subsection 5.1.2, the operating point of each compressor in the parallel configuration depends only on the speed applied to that compressor. Hence, the local input z_i for the i -th compressor is defined as

$$z_i = \omega_i. \quad (5.3)$$

Note that the local input z_i in (5.3) does not include interconnection variables. This is due to the fact that, in Problem (5.1), there are no interconnection constraints in the form of (4.5d). Nevertheless, Problem (5.1) has equality constraint (5.1b) that couples dynamics of the plant. Therefore, the gradient correction of each $m_{out,i}$ with respect to local inputs is required.

The load-sharing optimization problem at the k -th RTO iteration is handled via MA formulation:

$$\boldsymbol{\omega}_{k+1}^* = \underset{\boldsymbol{\omega}}{\operatorname{argmin}} \sum_{i \in \mathcal{N}} \Phi_i(\omega_i) + \left(\lambda_k^{\Phi_i} \right) (\omega_i - \omega_{i,k}) \quad (5.4a)$$

s.t. steady-state model equations

$$\mathcal{N}_{\mathcal{A}k}(\boldsymbol{\omega}) = \{i \in \mathcal{N} \mid G_{1,i}^a(\omega_i) + \varepsilon_k^{G_{1,i}} \geq 0, \text{ with } V_{rec,i}^a = 0\}, \quad (5.4b)$$

$$G_{1,j}(\omega_j) + \varepsilon_k^{G_{1,j}} = 0, \quad \forall j \in \mathcal{N}_{\mathcal{A}k}(\boldsymbol{\omega}), \quad (5.4c)$$

$$V_{rec,l} = 0, \quad \forall l \in \mathcal{N} \setminus \mathcal{N}_{\mathcal{A}k}(\boldsymbol{\omega}), \quad (5.4d)$$

$$\left(\sum_{i \in \mathcal{N}} m_{out,i}(\omega_i) + \left(\lambda_k^{m_i} \right)^\top (\omega_i - \omega_{i,k}) \right) + \varepsilon_k^m = m_{tot}^{sp}, \quad (5.4e)$$

inequality constraints (5.1e),

where the zeroth-order modifiers are computed as follows:

$$\varepsilon_k^m = m_{tot,p}(\boldsymbol{\omega}_k) - \sum_{i \in \mathcal{N}} m_{out,i}(\omega_{i,k}), \quad (5.5a)$$

$$\varepsilon_k^{G_{1,i}} = G_{1,p,i}(\omega_{i,k}) - G_{1,i}(\omega_{i,k}), \quad \forall i \in \mathcal{N}. \quad (5.5b)$$

Here, $m_{tot,p}(\boldsymbol{\omega}_k)$ is the measured value of the total mass flow at the k -th iteration and, for each compressor, $\lambda_k^{\Phi_i}$ and $\lambda_k^{m_i}$ are the first-order modifiers of $\Phi_{p,i}$ and $m_{out,p,i}$, respectively. $\mathcal{N}_{\mathcal{A}k}(\boldsymbol{\omega})$ is the set of compressors for which the surge constraint is active. The compressors that belong to this set are determined in (5.4b) by defining an auxiliary recycle valve opening that is always closed $V_{rec,j}^a = 0$, and an auxiliary surge control distance $-G_{1,j}^a$. The surge constraints that are violated for closed recycle valves are handled as active surge constraints in (5.4c). These equality constraints can be enforced by computing the required (predicted) valve opening $V_{rec,j}$, $\forall j \in \mathcal{N}_{\mathcal{A}k}(\boldsymbol{\omega})$. $\mathcal{N} \setminus \mathcal{N}_{\mathcal{A}k}(\boldsymbol{\omega})$ represents the complement of $\mathcal{N}_{\mathcal{A}k}(\boldsymbol{\omega})$, that is, the set of all compressors for which the surge control distance is strictly positive, i.e., $-G_{1,i}^a(\omega_i) - \varepsilon_k^{G_{1,i}} > 0$. For these compressors, the recycle valves are completely closed, as expressed in (5.4d). Note that condition (5.4b) introduces a discontinuity in the steady-state model equations, which describes the activation (or deactivation) of the anti-surge controller when the surge control distance becomes equal to zero. As a result, the steady-state compressor functions present a discontinuity in their derivatives with respect to the applied compressor speed. This discontinuity can be observed for the output mass flow, pressures and power consumption in Figure 5.2.

Note in addition that, (i) the gradient modifiers $\lambda_k^{\Phi_i}$ and $\lambda_k^{m_i}$ are scalars, (ii) the choke constraints have been eliminated to simplify the formulation, and (iii) in order to estimate the plant derivatives, a single perturbation of the speed ω_i is required for each compressor. Furthermore, it is possible to perturb all the speeds simultaneously in such a way that the disturbance to the total mass flow is negligible, as will be explained in Section 5.3.

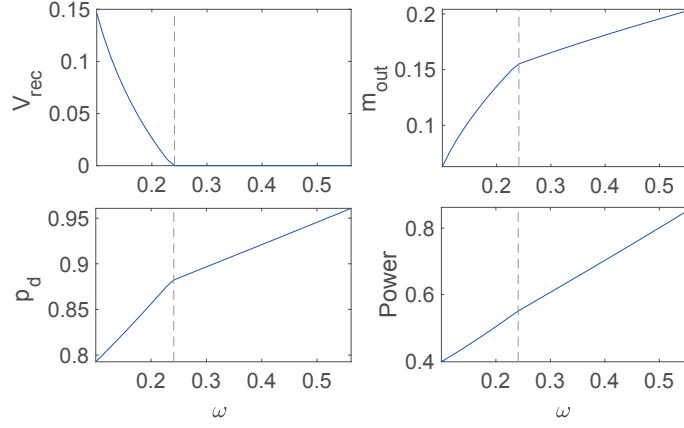


Figure 5.2 – Normalized values of recycle valve opening V_{rec} , discharge pressure p_d , output mass flow m_{out} and power consumption Φ as functions of speed ω . Discontinuity in the gradient is indicated with black dashed line.

5.2.2 MA for Serial Compressors

As already explained in Subsection 5.1.3, the outlet flow of each compressor is equal to

$$m_{out,N} = k_{out,N} V_{out,N} \sqrt{|p_{d,N} - p_{out,N}|}. \quad (5.6)$$

Consequently, if $p_{d,N}$, $p_{out,N}$ and $V_{out,N}$ do not change, the outlet flow of all compressors will remain constant at the plant steady state. Because of this, and due to the fact that the discharge of the i -th compressor feeds into the suction of the $(i + 1)$ -st compressor, the operating conditions of each compressor are defined by its speed ω_i and inlet pressure $p_{in,i}$. Hence, the local inputs z_i for the i -th compressor are defined as

$$\mathbf{z}_i = [\omega_i \quad p_{in,i}]^T. \quad (5.7)$$

In contrast to the parallel configuration, there is the interconnection variable $v_i = p_{in,i}$, which is involved in the interconnection constraint (5.2c), while the output is $y_i = p_{d,i}$.

Hence, Problem (5.2) can be addressed via the following MA formulation:

$$\left(\boldsymbol{\omega}_{k+1}^*, \mathbf{p}_{in,k+1}^*\right) = \underset{\boldsymbol{\omega}, \mathbf{p}_{in}}{\operatorname{argmin}} \sum_{i \in \mathcal{N}} \Phi_i(\mathbf{z}_i) + \left(\boldsymbol{\lambda}_k^{\Phi_i}\right)^\top (\mathbf{z}_i - \mathbf{z}_{p,i,k}) \quad (5.8a)$$

s.t. steady-state model equations

$$\mathcal{N}_{\mathcal{A}k}(\boldsymbol{\omega}) = \{i \in \mathcal{N} \mid G_{1,i}^a(\mathbf{z}_i) + \varepsilon_k^{G_{1,i}} \geq 0, \text{ with } V_{rec,i}^a = 0\}, \quad (5.8b)$$

$$G_{1,j}(\mathbf{z}_j) + \varepsilon_k^{G_{1,j}} = 0, \quad \forall j \in \mathcal{N}_{\mathcal{A}k}(\boldsymbol{\omega}), \quad (5.8c)$$

$$V_{rec,l} = 0, \quad \forall l \in \mathcal{N} \setminus \mathcal{N}_{\mathcal{A}k}(\boldsymbol{\omega}), \quad (5.8d)$$

$$p_{d,m,i} = p_{d,i} + \varepsilon_k^{p_{d,i}} + \left(\boldsymbol{\lambda}_k^{p_{d,i}}\right)^\top (\mathbf{z}_i - \mathbf{z}_{p,i,k}), \quad (5.8e)$$

$$p_{in,i+1} = p_{d,m,i} \quad i \in \{1, \dots, N-1\}, \quad (5.8f)$$

$$p_{d,m,N} = p_{d,N}^{\text{sp}}, \quad (5.8g)$$

$$p_{in,i} \in \mathcal{P}_{in,i}, \quad (5.8h)$$

inequality constraints (5.2g),

where the zeroth-order modifiers $\varepsilon_k^{p_{d,i}}$ are computed as:

$$\varepsilon_k^{p_{d,i}} = p_{d,p,i,k}(\mathbf{z}_{p,i,k}) - p_{d,i,k}(\mathbf{z}_{p,i,k}), \quad \forall i \in \mathcal{N}. \quad (5.9a)$$

Here, $\boldsymbol{\lambda}_k^{p_{d,i}}$ and $\boldsymbol{\lambda}_k^{\Phi_i}$ are the first-order modifiers of the plant cost $\Phi_{p,i}$ and plant discharge pressure $p_{d,p,i}$ with respect to \mathbf{z}_i , respectively. The zeroth- and first-order modifiers of $G_{1,i}(\mathbf{z}_i)$ are defined in the same way as in the previous subsection. Similar to the parallel case, the set of active surge constraints $\mathcal{N}_{\mathcal{A}k}(\boldsymbol{\omega})$ at the k -th RTO iteration and at speed $\boldsymbol{\omega}$ is determined by means of (5.8b).

Note that, (i) the gradient modifiers $\boldsymbol{\lambda}_k^{\Phi_i}$ and $\boldsymbol{\lambda}_k^{p_{d,i}}$ are vectors, (ii) the choke constraints have been dropped, and (iii) perturbations of the inputs \mathbf{z}_i are required to estimate the plant derivatives for each compressor. However, in this case it is also possible to perturb all the speeds simultaneously in such a way that the disturbance of the outlet pressure setpoint $p_{d,N}^{\text{sp}}$ is negligible, as will be explained in Section 5.3.

5.2.3 Implementation Aspects

As shown in Figure 5.1, the discharge pressure $p_{d,N}$ and the mass flow m_{tot} are regulated according to the station demands $p_{d,N}^{\text{sp}}$ and m_{tot}^{sp} for the serial and parallel configurations, respectively. In conventional load sharing, each compressor receives the same speed setpoint $\omega_k^c(t)$, thereby achieving equal-load distribution among the interconnected compressors. Load-sharing optimization comes as an additional layer that generates asymmetries in the load distribution based on the current efficiencies of the various units.

Consider the steady state reached at the k -th RTO iteration, with ω_k^c the speed setpoint generated by the mass-flow or pressure controller and $\omega_{i,k}$ the speed setpoint applied to the i -

th compressor. The RTO algorithm suggests applying next $\omega_{i,k+1}^f$, which is the filtered value of the optimal speed computed from Problem (5.4) or Problem (5.8) for parallel and serial configuration, respectively, that is:

$$\omega_{i,k+1}^f = K\omega_{i,k+1}^* + (I - K)\omega_{i,k}, \quad (5.10)$$

which means that the contribution of the optimization to the next RTO iteration is:

$$\Delta\omega_{i,k+1}^f = \omega_{i,k+1}^f - \omega_k^c. \quad (5.11)$$

This closed-loop implementation maintains the attractive MA property of converging to a Karush-Kuhn-Tucker (KKT) point for the plant (see Theorem 1 in Marchetti et al. (2009)).

5.3 Gradient Estimation Exploiting the Problem Structure

The most challenging part of MA consists in estimating the plant gradients or the first-order modifiers used in (5.4) and (5.8). In general, steady-state perturbation methods require at least $(n_u + 1)$ steady-state operating points to be available, which means that one has to wait for steady state each time the inputs are changed. It turns out that MA for interconnected systems can help since the input dimension of each compressor is $n_{z_i} = 2$ in serial, and $n_{z_i} = 1$ in parallel configuration, respectively. The price for being able to reduce the dimensionality of the input space via MA for interconnected systems is that the plant gradients with respect to the interconnection variables are needed.

We show next that, in the case of the load-sharing Problems (5.4) and (5.8) the plant structure can be exploited for estimating the plant derivatives efficiently. In fact, it is possible to perturb the speeds of all compressors at the same time to estimate the required gradients. This has the advantage that the complete gradients can be estimated using only two steady-state operating points, regardless of the number of compressors. This will be explained in the following subsections. These steady states correspond to the current RTO operating point ω_k and the perturbed operating point $\tilde{\omega}_k$. The perturbed point is obtained by adding the perturbations $\Delta\omega_{i,k}$ to each compressor (see Figure 5.1):

$$\tilde{\omega}_{i,k} = \tilde{\omega}_k^c + \Delta\omega_{i,k}^f + \Delta\omega_{i,k}, \quad (5.12)$$

where $\tilde{\omega}_k^c$ is the speed generated by the mass-flow or pressure controller at steady state for the perturbed operating point. The speed perturbation vector $\Delta\omega_k$ is defined as:

$$\Delta\omega_k = \text{diag}(\Delta\omega_k) \mathbf{d}_k, \quad (5.13)$$

where $\mathbf{d}_k \in \mathbb{R}^N$ is a vector whose i -th element $d_{i,k}$ can take only the values 1 or -1 , defining the direction of the perturbation for the i -th compressor; and $\Delta\omega_k \in \mathbb{R}^N$ represents the size of the perturbation for the compressor speeds. The choices of \mathbf{d}_k and $\Delta\omega_k$ are discussed in the

following subsections.

5.3.1 Direction of Perturbations

The directions of the perturbations will be selected based on an heuristic approach that leads to good gradient estimates in practice. Since we want to perturb all compressors at the same time without significantly perturbing the setpoint-matching conditions (5.4e) or (5.8g), we choose to perturb half of the compressors in one direction and the other half in the opposite direction.

Due to the update of the surge constraints in (5.4c) and (5.8c), the modified model provides good estimates of the speed values at which the discontinuities occur. Thus, the directions of the perturbations will be selected based on the proximity to the discontinuity. If N is even, then the $N/2$ compressors closest to the discontinuity point are assigned directions that push them away from the discontinuity point. The other compressors are assigned directions such that $\sum_{i \in \mathcal{N}} d_{i,k} = 0$. If N is odd, then the $(N + 1)/2$ compressors closest to the discontinuity point are assigned directions that push them away from the discontinuity point. The other compressors are assigned directions such that $\sum_{i \in \mathcal{N}} d_{i,k} = \pm 1$.

5.3.2 Gradient Estimation for Parallel Configuration

The derivatives of the power consumption are estimated by finite difference as follows :

$$\widehat{\nabla_{\omega_i} \Phi_{p,i}}(\omega_{i,k}) = \frac{\Phi_{p,i}(\tilde{\omega}_{i,k}) - \Phi_{p,i}(\omega_{i,k})}{\tilde{\omega}_{i,k} - \omega_{i,k}}. \quad (5.14)$$

The mass-flow derivatives $\widehat{\nabla_{\omega_i} m_{c,i}}$ can be estimated using the same approach. An operating line of the compressor in parallel connection is shown in red in Figure 5.3. The perturbations $\Delta z_{i,k} = \Delta \omega_{i,k}$ should be selected such that (i) gradients are estimated with sufficient accuracy, (ii) the perturbation of the total mass flow is negligible.

These two requirements can be accounted for by computing the perturbation steps Δ_{ω_k} in (5.13), for the given direction vector \mathbf{d}_k , by solving the following optimization problem at the

k -th RTO iteration:

$$\Delta_{\omega_k} = \underset{\Delta_{\omega}}{\operatorname{argmin}} \sum_{i \in \mathcal{N}} \mathcal{E}^{L_i}(\tilde{\omega}_i) \quad (5.15a)$$

s.t. steady-state equations

$$\mathcal{N}_{\mathcal{A}k}(\tilde{\omega}) = \{i \in \mathcal{N} \mid G_{1,i}^a(\tilde{\omega}_i) + \varepsilon_k^{G_{1,i}} \geq 0, \text{ with } V_{rec,i}^a = 0\}, \quad (5.15b)$$

$$G_{1,j}(\tilde{\omega}_j) + \varepsilon_k^{G_{1,j}} = 0, \quad \forall j \in \mathcal{N}_{\mathcal{A}k}(\tilde{\omega}), \quad (5.15c)$$

$$V_{rec,l} = 0, \quad \forall l \in \mathcal{N} \setminus \mathcal{N}_{\mathcal{A}k}(\tilde{\omega}), \quad (5.15d)$$

$$\tilde{\omega} = \omega_k + \operatorname{diag}(\Delta_{\omega}) \mathbf{d}_k, \quad (5.15e)$$

$$\left(\sum_{i \in \mathcal{N}} m_{out,i}(\tilde{\omega}_i) + (\lambda_{k-1}^{m_i})^\top (\tilde{\omega}_i - \omega_{i,k}) \right) + \varepsilon_k^m = m_{tot}^{sp}, \quad (5.15f)$$

$$0 \leq \Delta_{\omega_i} \leq \Delta_{\omega_i}^U, \quad i \in \mathcal{N} \quad (5.15g)$$

where $\tilde{\omega}$ is the optimal perturbed speed of optimization problem (5.15) and $\mathcal{E}^{L_i}(\tilde{\omega}_i)$ is the bound on gradient error function

$$\mathcal{E}^{L_{p,i}}(\tilde{\omega}_i) = \mathcal{E}^{\Phi_{p,i}} + \|\mu_{m_{out,i}}\| \mathcal{E}^{m_{out,p,i}} \quad (5.16)$$

where the value of the Lagrange multiplier $\mu_{m_{out,i}}$ is obtained at the optimum of the modified model at $(k-1)$ -th iteration. Note that the gradient modifiers $\lambda_{k-1}^{m_i}$ used in Problem (5.15) are the ones corresponding to the $(k-1)$ -th RTO iteration, since Problem (5.15) is solved to compute the perturbations required for computing the gradient modifiers $\lambda_k^{\Phi_i}$ and $\lambda_k^{m_i}$.

The accuracy in the gradient estimates in (i) is enforced by selecting a sufficiently large perturbation step Δ_{ω_i} . The requirement (ii) is enforced by including the equality constraint (5.15f).

5.3.3 Gradient Estimation for Serial Configuration

For the serial configuration, we need to estimate the gradients of the cost and constraints with respect to both ω_i and $p_{in,i}$. While the compressor speeds can be conveniently perturbed, the difficulty in estimating the gradients comes from the fact that the inlet pressures (interconnection variables) cannot be independently perturbed. Next, we describe how the gradients can be estimated by perturbing only the compressor speeds.

In perturbing a compressor's speed, we can estimate the plant gradients along the direction denoted $\mathbf{U}_{i,1}^r = \tilde{\mathbf{z}}_{p,i,k} - \mathbf{z}_{p,i,k}$, where $\tilde{\mathbf{z}}_{p,i,k} = [\tilde{\omega}_{i,k} \quad \tilde{p}_{in,p,i,k}]^\top$ with $\tilde{p}_{in,p,i,k}$ being the measured inlet pressure at the perturbed speed $\tilde{\omega}_k$ and $\mathbf{z}_{p,i,k} = [\omega_{i,k} \quad p_{in,p,i,k}]^\top$.

The blue line in Figure 5.3 corresponds to the operating line of a compressor in serial connection. A compressor in serial connection operates at constant outlet mass flow $m_{out,i}$, and thus, when the recycle valve is closed the compressor mass flow $m_{c,i}$ is also constant

5.3. Gradient Estimation Exploiting the Problem Structure

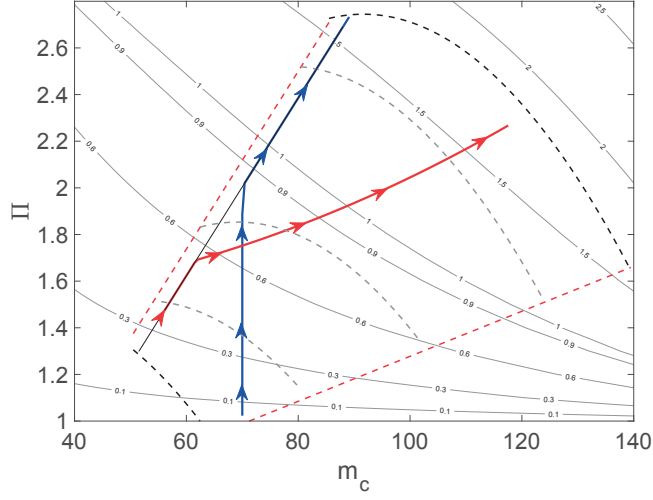


Figure 5.3 – Compressor operating constraints and power consumption contour curves (continuous gray line) in the Π - m_c space. Upper and lower bound on the speed (dashed black); operating points for three constant speeds (dashed gray); choke and surge line (dashed red); and surge control line parallel to surge line (black). Operating steady-state points for a compressor in parallel configuration (red line) and operating points for a compressor in serial configuration (blue line). The arrows point in the direction of speed increase.

at $m_{c,i} = m_{out,i}$. One can notice in Figure 5.3 that a dashed gray curve corresponds to operating conditions where the compressor speed is constant. The intersection of these gray curves with the blue operating line correspond to operating points for different speed values. Hence, operating at constant outlet mass flow and constant speed is equivalent to operating on a fixed point on the blue operating line. For any fixed operating point on the blue line, the power consumption and the pressure ratio are also fixed. However, the inlet pressure might vary since it is affected by upstream compressor. Based on this analysis, the power consumption gradient can be estimated by selecting as the second direction $\mathbf{U}_{i,2}^r = [0 \ 1]^\top$, that is, the direction where ω_i is constant and $p_{in,i}$ changes, and setting the variation of the power consumption along this direction to zero

$$\widehat{\nabla_{\mathbf{z}_i} \Phi_{p,i}}^\top = \begin{bmatrix} \Phi_{p,i}(\bar{\mathbf{z}}_{p,i,k}) - \Phi_{p,i}(\mathbf{z}_{p,i,k}) \\ 0 \end{bmatrix}^\top \mathbf{U}_{i,1}^{-1}, \quad (5.17)$$

where $\mathbf{U}_{i,1} = [\mathbf{U}_{i,1}^r \ \mathbf{U}_{i,2}^r]$.

For the estimation of $\nabla_{\mathbf{z}_{p,i}} p_{d,p,i}$ in (5.8), we consider the discharge pressure in (2.22c) at steady state, $p_{d,p,i} = \Pi_{p,i} p_{s,p,i}$, and write

$$\widehat{\nabla_{\mathbf{z}_{p,i}} p_{d,p,i}} = \widehat{\nabla_{\mathbf{z}_{p,i}} \Pi_{p,i}} p_{s,p,i} + \Pi_{p,i} \widehat{\nabla_{\mathbf{z}_{p,i}} p_{s,p,i}}, \quad (5.18a)$$

Chapter 5. Application to Gas-Compressor Stations

where, similarly to the power consumption in (5.17), we estimate $\nabla_{\mathbf{z}_i} \Pi_{p,i}$ as

$$\widehat{\nabla_{\mathbf{z}_{p,i}} \Pi_{p,i}}^\top = \begin{bmatrix} \Pi_{p,i}(\tilde{\mathbf{z}}_{p,i,k}) - \Pi_{p,i}(\mathbf{z}_{p,i,k}) \\ 0 \end{bmatrix}^\top \mathbf{U}_{i,1}^{-1}. \quad (5.18b)$$

For the estimate of $\widehat{\nabla_{\mathbf{z}_{p,i}} p_{s,p,i}}$, we use the fact that the suction pressure does not change for the constant inlet/outlet mass flow and constant inlet pressure according to (2.23a), which corresponds to the direction $\mathbf{U}_{i,3}^r = [1 \ 0]^\top$. The gradient estimate of the suction pressure is obtained as

$$\widehat{\nabla_{\mathbf{z}_{p,i}} p_{s,p,i}}^\top = \begin{bmatrix} p_{s,p,i}(\tilde{\mathbf{z}}_{p,i,k}) - p_{s,p,i}(\mathbf{z}_{p,i,k}) \\ 0 \end{bmatrix}^\top \mathbf{U}_{i,2}^{-1}, \quad (5.18c)$$

where $\mathbf{U}_{i,2} = [\mathbf{U}_{i,1}^r \ \mathbf{U}_{i,3}^r]$. The perturbations $\Delta\omega_{i,k}$ are selected in a similar way as we proposed for parallel compressors. At the k -th RTO iteration, and for a given direction vector \mathbf{d}_k , the perturbation steps $\Delta\omega_k$ are obtained by solving the following optimization problem:

$$\left(\Delta\omega_k, \Delta p_{in,k} \right) = \underset{\Delta\omega, \Delta p_{in}}{\operatorname{argmin}} \sum_{i \in \mathcal{N}} \mathcal{E}^{L,p,i}(\tilde{\mathbf{z}}_i) \quad (5.19a)$$

s.t. steady-state equations

$$\tilde{\mathbf{z}}_i = [\tilde{\omega}_i \ \tilde{p}_{in,i}]^\top \quad (5.19b)$$

$$\tilde{\mathbf{p}}_{in} = \mathbf{p}_{in,p,k} + \Delta p_{in}, \quad (5.19c)$$

$$\tilde{\omega} = \omega_k + \operatorname{diag}(\Delta\omega) \mathbf{d}_k, \quad (5.19d)$$

$$\mathcal{N}_{\mathcal{A}k}(\tilde{\omega}) = \{i \in \mathcal{N} \mid G_{1,i}^a(\tilde{\mathbf{z}}_i) + \varepsilon_k^{G_{1,i}} \geq 0, \text{ with } V_{rec,i}^a = 0\}, \quad (5.19e)$$

$$G_{1,j}(\tilde{\mathbf{z}}_j) + \varepsilon_k^{G_{1,j}} = 0, \forall j \in \mathcal{N}_{\mathcal{A}k}(\tilde{\omega}), \quad (5.19f)$$

$$V_{rec,l} = 0, \forall l \in \mathcal{N} \setminus \mathcal{N}_{\mathcal{A}k}(\tilde{\omega}), \quad (5.19g)$$

$$p_{d,m,i} = p_{d,i} + \varepsilon_k^{p_{d,i}} + (\boldsymbol{\lambda}_{k-1}^{p_{d,i}})^\top (\tilde{\mathbf{z}}_i - \mathbf{z}_{p,i,k}), \quad (5.19h)$$

$$\tilde{p}_{in,i+1} = p_{d,m,i} \quad i \in \{1, \dots, N-1\}, \quad (5.19i)$$

$$p_{d,m,N} = p_{d,N}^{\text{sp}}, \quad (5.19j)$$

$$\tilde{p}_{in,i} \in \mathcal{P}_{in,i}, \quad (5.19k)$$

$$0 \leq \Delta\omega_i \leq \Delta\omega_i^U, \quad i \in \mathcal{N} \quad (5.19l)$$

$$\Delta_{p_{in,i}}^L \leq \Delta p_{in,i} \leq \Delta_{p_{in,i}}^U, \quad i \in \mathcal{N}. \quad (5.19m)$$

Here, $\mathbf{p}_{in,p,k}$ is the measured inlet pressure; $\tilde{\omega}$ is the optimal perturbed speed of optimization problem (5.19), while $\tilde{\mathbf{p}}_{in}$ is the optimal perturbed inlet pressure at $\tilde{\omega}$. Moreover, $\mathcal{E}^{L,p,i}(\tilde{\mathbf{z}}_i)$ in (5.19a) is given by

$$\mathcal{E}^{L,p,i}(\tilde{\mathbf{z}}_i) = \mathcal{E}^{\Phi,p,i}(\tilde{\mathbf{z}}_i) + \|\mu_{p_{d,i}}\| \mathcal{E}^{p_{d,p,i}}(\tilde{\mathbf{z}}_i) \quad (5.19n)$$

where the value of the Lagrange multiplier $\mu_{p_{d,i}}$ is obtained at the optimum of the modified model at $(k-1)$ -th iteration; $\mathcal{E}^{\Phi,p,i}(\tilde{\mathbf{z}}_i)$ and $\mathcal{E}^{p_{d,p,i}}(\tilde{\mathbf{z}}_i)$ are individual bound on error functions

5.3. Gradient Estimation Exploiting the Problem Structure

of the cost and discharge pressure, respectively. Note that the analysis of Subsection 2.2.4 assumes that upper bounds on gradient error due to measurement noise and truncation are defined with the same input matrix \mathbf{U}_i . However, different input matrices $\mathbf{U}_{i,1}$ and $\mathbf{U}_{i,2}$ are utilized for obtaining $\widehat{\nabla_{\mathbf{z}_{p,i}} p_{d,p,i}}$ in (5.18a). Thus $\mathcal{E}^{p_{d,p,i}}(\tilde{\mathbf{z}}_i)$ is given by

$$\mathcal{E}^{p_{d,p,i}}(\tilde{\mathbf{z}}_i) = \|p_{s,p,i}(\mathbf{z}_{p,i,k})\| \mathcal{E}^{\Pi_{p,i}}(\tilde{\mathbf{z}}_i) + \|\Pi_{p,i}(\mathbf{z}_{p,i,k})\| \mathcal{E}^{p_{s,p,i}}(\tilde{\mathbf{z}}_i), \quad (5.19o)$$

where $\mathcal{E}^{\Pi_{p,i}}(\tilde{\mathbf{z}}_i)$ and $\mathcal{E}^{p_{s,p,i}}(\tilde{\mathbf{z}}_i)$ are individual upper bounds on error functions of the $\Pi_{p,i}$ and $p_{s,p,i}$.

5.3.4 Estimation of Discontinuous Compressor Function Gradients

A difficulty in estimating experimental gradients in the neighborhood of the surge control line is due to the discontinuity in the gradients. The reason why and when this discontinuity occurs is illustrated in Figures 5.3 and 5.4. Figure 5.3 depicts two operating curves. By observing the red curve that corresponds to the operating line for compressor in the parallel configuration, one sees that, at low speeds, the compressor operates on the surge control line, that is, the anti-surge controller is activated to avoid potentially dangerous conditions. The point where the surge line passes from being active to inactive (or vice versa) is the point of discontinuity. At this point, the derivatives of the compressor functions, such as mass flow, power consumption and pressures, present a discontinuity.

This discontinuity is illustrated for the outlet mass-flow function $m_{out,i}$ in Figure 5.4. The surge constraint $G_{1,i}$ is also depicted in Figure 5.4. The shaded area in both plots indicates the region where the plant compressor is on the surge control line, that is, when the plant function $G_{1,p}$ (depicted in blue) is active with $G_{1,p,i} = 0$ for all speed values less than $\omega_i = 0.14$. One also notices that the slope of the outlet mass-flow function $m_{out,p,i}$ is higher on the surge line. Due to plant-model mismatch, the model surge constraint $G_{1,i}$ predicts this discontinuity at a much higher speed (around $\omega_i = 0.24$).

This results in high mismatch between the plant (blue) and the model (black) outlet mass-flow functions. Hence, we may conclude that, besides having different slopes, the outlet mass-flow functions have different points of discontinuity. We remedy this issue by adding zeroth-order modifiers to G_1 . Adding modifiers to the surge function G_1 in (5.4d) and (5.8d) results in a good prediction of the point where the discontinuity occurs. Due to this adaptation, we obtain the new output mass-flow function (dashed black line). The modified function $m_{out,i}$ (red line) is obtained by adding the zeroth- and first-order modifiers.

On the one hand, with the approach of estimating modifiers via finite-difference in (2.16), the modified cost and constraints are such that their values match the corresponding measured values at the current operating point $\mathbf{z}_{p,i,k}$, and their gradients match the corresponding gradient estimates at $\mathbf{z}_{p,i,k}$. On the other hand, with the modifier estimation approach based on linear regression in (2.21) and further discussed for interconnected systems in (4.15b),

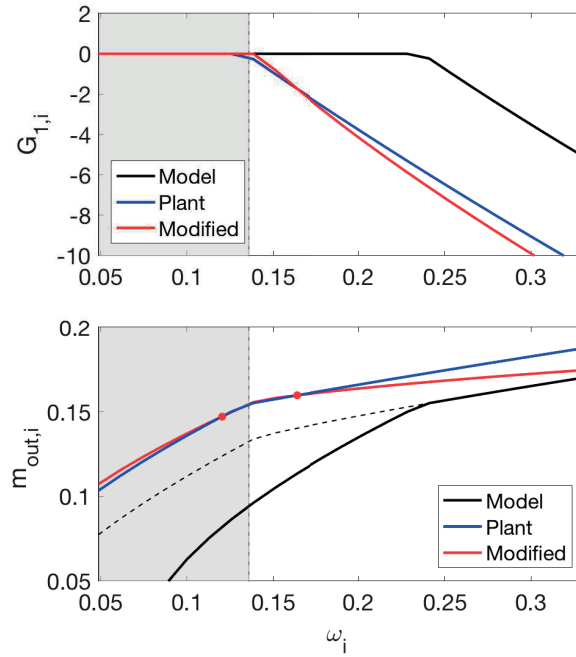


Figure 5.4 – **Top plot:** Surge constraint as a function of the speed. The gray area represents the region where the surge constraint is active, $G_{1,p,i} = 0$. **Bottom plot:** Output mass flow as a function of the speed. The two red points represent the measured plant outlet flow at the operating point $\omega_{i,k}$ and the perturbed point $\tilde{\omega}_{i,k}$. The dashed black curve is the adapted model outlet flow following the adaptation of the surge constraint $G_{1,i}$.

the modified cost and constraint functions match the corresponding measured values for the plant at the current and perturbed operating points. This gives a better approximation of the plant cost and constraint functions, in particular for increased distances between the points.

The choice between the modifiers in (2.16) and (2.21) greatly affects the convergence of the MA schemes in Section 5.2.1 and Section 5.2.2. In fact, when the compressor operates on the surge line, the first-order correction of the slope given by the modifiers in (2.16) would adversely affect the function slope in non-surge region and vice versa.

5.4 Simulation Results

The simulations used to generate the results are based on industrial compressor models and were implemented in Matlab/Simulink. Plant-model mismatch is introduced by using compressor maps (2.24) that are different for the plant and the model. We consider the scenario of a gas compressor station consisting of three compressors ($N = 3$) for both parallel and serial configuration. The plant efficiency maps A, B and C shown in Figure 5.5 are used in (2.25a) to compute the plant cost functions $\Phi_{p,i}$. The model cost functions Φ_i are obtained using the model efficiency map shown in Figure 5.5. The model assumes the same map for all com-

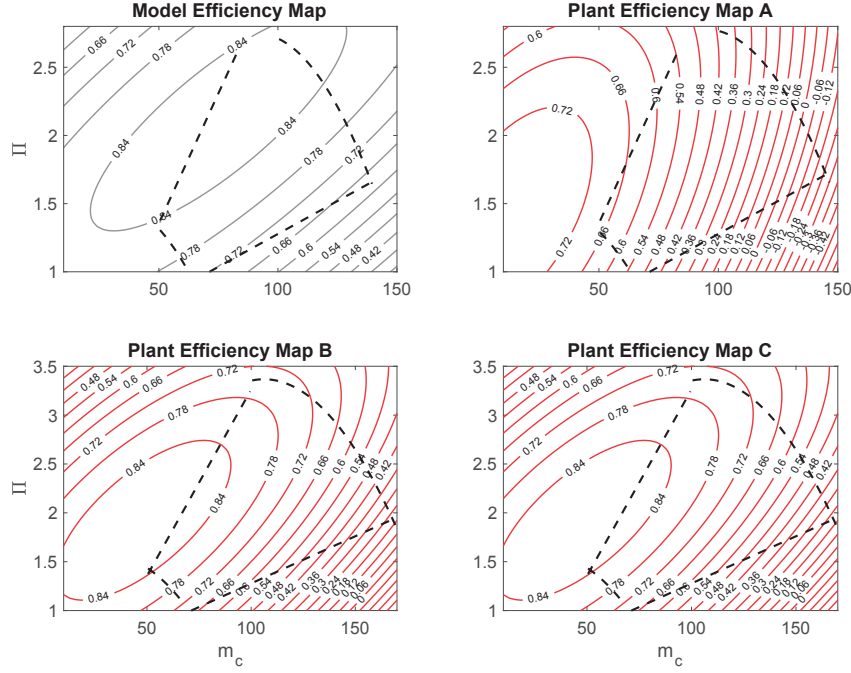


Figure 5.5 – Comparison between the model efficiency map (grey) and the plant efficiency maps (red) for three different compressors. The maps are shown as contour plots, whereas the dashed black lines represent the operating range of the gas compressor.

pressors. It follows that the model optimum corresponds to equal-load distribution for both the parallel and serial configurations.

5.4.1 Parallel Configuration

The MA scheme described in Subsection 5.2.1 is tested in simulation. Once the controlled plant satisfies near steady-state conditions, cost and constraint measurements are taken and averaged over a moving time window of 10 s.

The perturbations $\Delta\omega_{i,k}$ used to estimate the gradients are computed according to the procedure discussed in Subsection 5.3.2. Based on experimental data from real compressors, the power consumption and the mass flow are selected as Gaussian noises with standard deviations of $\sigma_{\Phi_i} = 0.1$ and $\sigma_{m_{out,i}} = 0.004$, respectively. The noise interval is $\delta_{\Phi_i} = 6\sigma_{\Phi_i}$ and $\delta_{m_{out,i}} = 6\sigma_{m_{out,i}}$. Inputs z_i are scaled in the interval $[0,2]$. In this region, the largest eigenvalue of the Hessian of the cost and outlet mass flow predicted by the modified model, obtained with the scaled inputs, are $d_{\sigma_i}^{\Phi} = 6$, $d_{\sigma_i}^{m_{out}} = 3$, respectively. Thus $d_{\sigma_i}^{\Phi}$ and δ_{Φ_i} are used to obtain \mathcal{E}^{Φ_i} ; $d_{\sigma_i}^{m_{out}}$ and $\delta_{m_{out,i}}$ are used to obtain $\mathcal{E}^{m_{out,i}}$ in (5.16). The filter (5.10) is applied with $K = 0.6I$. Simulations start from the model optimum, that is, equal-load distribution.

Instead of perturbing the system at every RTO iteration, the perturbations are stopped

Chapter 5. Application to Gas-Compressor Stations

and the gradient modifiers no longer updated once the cost improvement becomes negligible. For instance, at the k -th iteration, excitation is not carried out if, for a given $\varepsilon > 0$, the following condition is met:

$$\left| \frac{\sum_{i=1}^3 \Phi_{p,i}(\mathbf{z}_{p,i,k}) - \sum_{i=1}^3 \Phi_{p,i}(\mathbf{z}_{p,i,k-1})}{\sum_{i=1}^3 \Phi_{p,i}(\mathbf{z}_{p,i,k-1})} \right| \leq \varepsilon. \quad (5.20)$$

Gradient estimation is restarted every time the total mass-flow setpoint is changed. To obtain smooth transitions between steady states, the changes in both the mass-flow setpoint and the feedforward contributions to the speeds are implemented using ramps.

Using the proposed MA scheme, the time profiles of the speeds applied to three compressors are shown in Figure 5.6. They are compared to the equal load distribution (model optimum) speeds obtained for the same station flow set-point variations. Also, time profiles of the surge distances are depicted in Figure 5.7. One sees that, when the compressors operate close to the surge conditions, the performed perturbations are in agreement with analysis presented in Section 5.3.1. The plant station mass flow $m_{tot,p}$ is compared to its setpoint for four different setpoint values in Figure 5.8. Note that the station mass flow satisfies the desired setpoint even when the speeds are being perturbed for the purpose of gradient estimation. Using $\varepsilon = 10^{-3}$ in (5.20), the perturbations are stopped after only 2-3 RTO iterations. As can be seen in Figure 5.8, the improvement in power consumption goes up to 8%, which is economically significant. Power loss in Figure 5.8 is obtained as

$$P_{loss} = \frac{\sum_{i=1}^3 \Phi_{p,i}(\mathbf{z}_{p,i,k}) - \sum_{i=1}^3 \Phi_{p,i}^*}{\sum_{i=1}^3 \Phi_{p,i}^*} \times 100\% \quad (5.21)$$

where $\Phi_{p,i}^*$ is the optimal power consumption of i -th compressor for the current output mass-flow setpoint value. Similarly, the instantaneous optimality loss after each pressure setpoint change is nearly suppressed after the first RTO iteration.

5.4.2 Serial Configuration

The MA scheme described in Subsection 5.2.2 is tested in simulation. The perturbations $\Delta\omega_{i,k}$ used to estimate the gradients are computed according to the procedure discussed in Subsection 5.3.3.

As already mentioned in Subsection 5.3.3, input matrices $\mathbf{U}_{i,1}$ and $\mathbf{U}_{i,2}$ are used to obtain the gradient estimates of $\hat{\nabla}_{\mathbf{z}_i} \Phi_{p,i}$ and $\nabla_{\mathbf{z}_i} p_{d,p,i}$ in (5.17), (5.18b) and (5.18c). These matrices are ill-conditioned if for the new perturbation point $\tilde{\omega}$, vector $\mathbf{U}_{i,1}^r = \tilde{\mathbf{z}}_{p,i,k} - \mathbf{z}_{p,i,k}$ lies in direction $\mathbf{U}_{i,2}^r$ or $\mathbf{U}_{i,3}^r$. Therefore, utilization of the upper bound on gradient error function $\mathcal{E}^{L_i}(\tilde{\mathbf{z}}_i)$ in (5.19n) penalizes the perturbations in these directions and provides a good candidate $\tilde{\omega}$ for accurate gradient estimation. Figure 5.9 depicts the contour lines of $\mathcal{E}^{L_i}(\tilde{\mathbf{z}}_i)$ function. One can notice in Figure 5.9 that values of $\mathcal{E}^{L_i}(\tilde{\mathbf{z}}_i)$ are very large in direction $\mathbf{U}_{i,2}^r$ and $\mathbf{U}_{i,3}^r$. Thus,

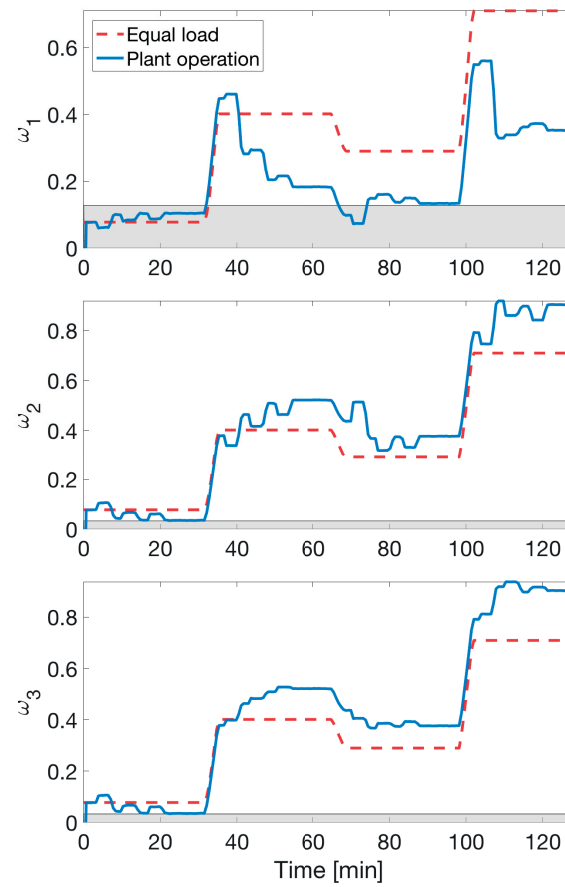


Figure 5.6 – Time profiles of the normalized speeds for parallel configuration. Applied speed using MA scheme is depicted in blue. Steady-state equal-load distribution speed is presented with red dashed line. The shaded area indicates the region where the compressors are operating on the surge control line.

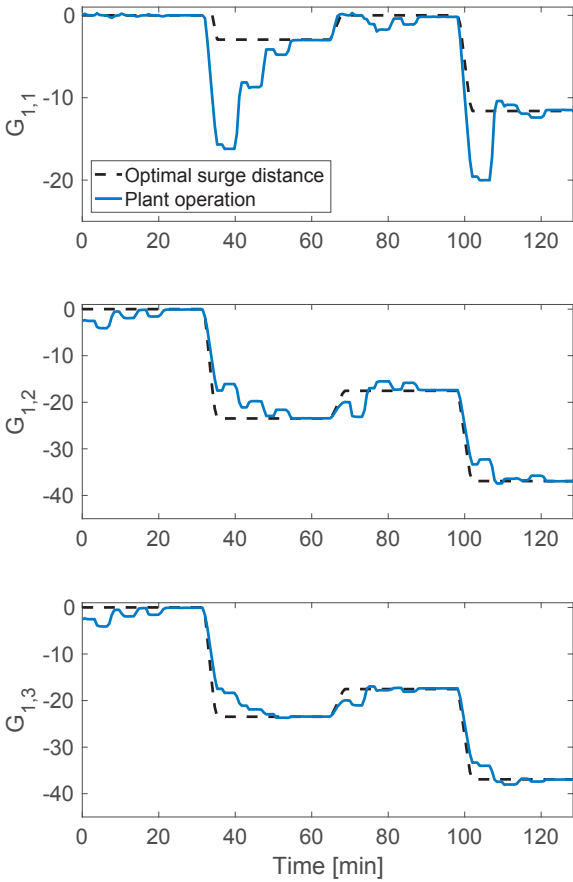


Figure 5.7 – Values of the plant surge control distance for parallel configuration (blue line). Surge distance corresponding to the plant optimum (black dashed line).

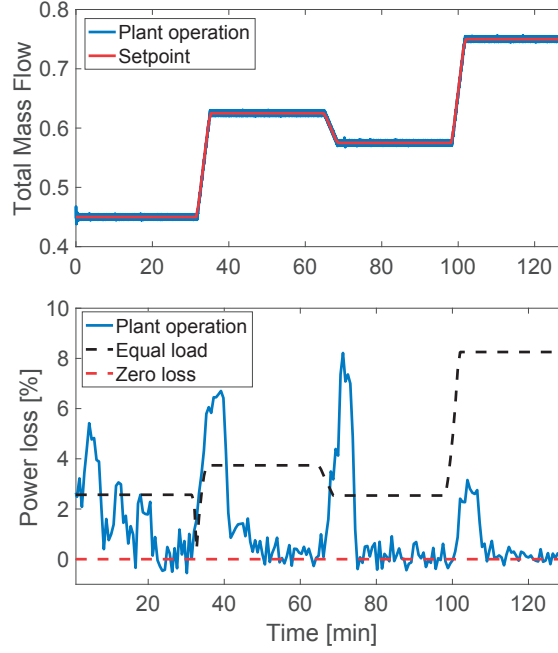


Figure 5.8 – **Top plot:** Normalized total station mass flow $m_{tot,p}$. **Bottom plot:** Power loss using MA compared to equal-load distribution. The plant optimum corresponds to zero power loss.

optimal perturbations $\Delta\omega_k, \Delta p_{in,k}$ will tend to be in one of the two valleys, illustrated in Figure 5.9, such that constraints in (5.19) are satisfied.

The measurement noises for the power consumption, discharge pressure and pressure ratio are selected as Gaussian noises with standard deviations of $\sigma_{\Phi_i} = 0.1$, $\sigma_{p_{s,p,i}} = 0.01$ and $\sigma_{\Pi_{p,i}} = 0.01$, respectively. The noise interval for each one of them is defined as $\delta_{\Phi_i} = 6\sigma_{\Phi_i}$, $\delta_{p_{s,p,i}} = 6\sigma_{p_{s,p,i}}$ and $\delta_{\Pi_{p,i}} = 6\sigma_{\Pi_{p,i}}$. Inputs \mathbf{z}_i are scaled in the interval $[0, 2]$. In this region, the largest eigenvalue of the Hessian of the cost, discharge pressure and pressure ration predicted by the modified model, obtained with the scaled inputs, are $d_{\sigma_i}^{\Phi} = 6$, $d_{\sigma_i}^{p_s} = 4$ and $d_{\sigma_i}^{\Pi} = 2$, respectively. Thus $d_{\sigma_i}^{\Phi}$ and δ_{Φ_i} are used to obtain \mathcal{E}^{Φ_i} ; $d_{\sigma_i}^{p_s}$ and $\delta_{p_{s,p,i}}$ are used to obtain $\mathcal{E}^{p_{s,p,i}}$; and $d_{\sigma_i}^{\Pi}$ and $\delta_{\Pi_{p,i}}$ are used to obtain $\mathcal{E}^{\Pi_{p,i}}$ in (5.19o) and (5.19n).

The filter (5.10) is applied with $K = 0.9I$. The filter gain K is larger here, in comparison to the parallel configuration, because the adaptation of the modifiers and thus the input values ω_k has been less aggressive for the chosen operating setpoints according to our findings. Simulations are run starting from model optimum, that is, equal-load distribution.

Using the proposed MA scheme, the time profiles of the speeds applied to the three compressors are shown in Figure 5.10. They are compared to the equal load distribution (model optimum) speeds obtained for the same station flow set-point variations. Also, the time profiles of the surge distances are depicted in Figure 5.11. In Figure 5.10, one may see that the

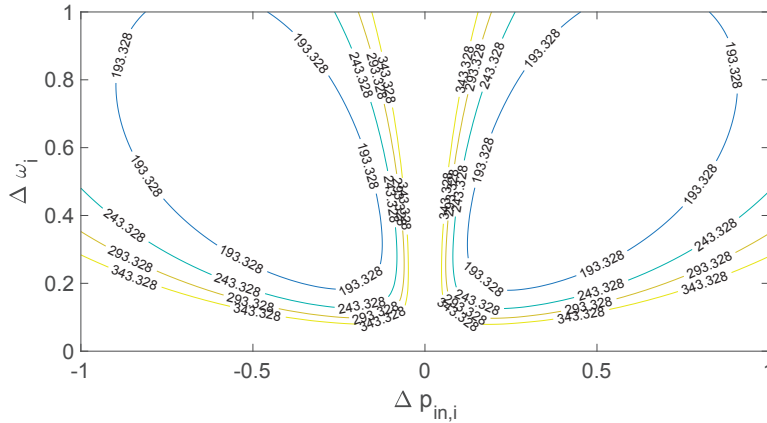


Figure 5.9 – Gradient error contour lines of one compressor for the serial connection for normalised perturbations $\Delta\omega_i$ and $\Delta p_{in,i}$.

performed perturbations are in agreement with the methodology presented in Section 5.3.1. This is clearly noticeable for the second and third compressors between 40 min - 60 min of operation. Namely, around 45 min, compressors operate on the surge line and perturbations are pushing them away from the discontinuity in order to stay in the surge conditions after perturbation. Then, around 50 min, RTO brings compressors out of the surge and perturbations are performed such that again compressors are pushed away from the discontinuity.

The plant station pressure $p_{d,3}$ is compared to its setpoints for four different setpoint values in Figure 5.12. Note that the station mass flow satisfies the desired setpoint even when the speeds are being perturbed for the purpose of gradient estimation. Using $\varepsilon = 10^{-3}$ in (5.20), the perturbations are stopped after only 2-3 RTO iterations. Figure 5.12 shows an improvement in power consumption of up to 10% for the simulated operating conditions. Power loss in Figure 5.12 is obtained as in (5.21). Similarly to the parallel configuration, the instantaneous optimality loss after each pressure setpoint change is nearly suppressed after the first RTO iteration.

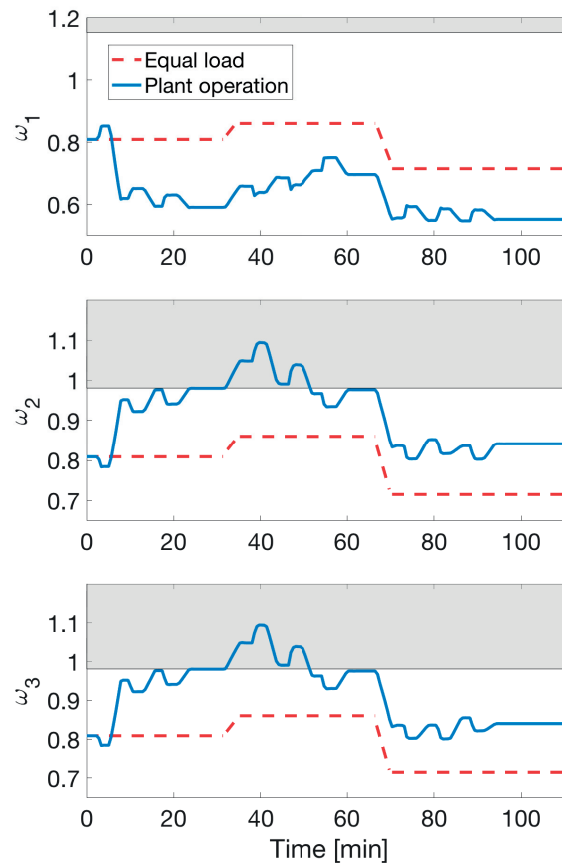


Figure 5.10 – Time profiles of the normalized speeds for serial configuration. Applied speed using MA scheme is depicted in blue. Steady-state equal-load distribution speed is presented with red dashed line. The shaded area indicates the region where the compressors are operating on the surge control line.

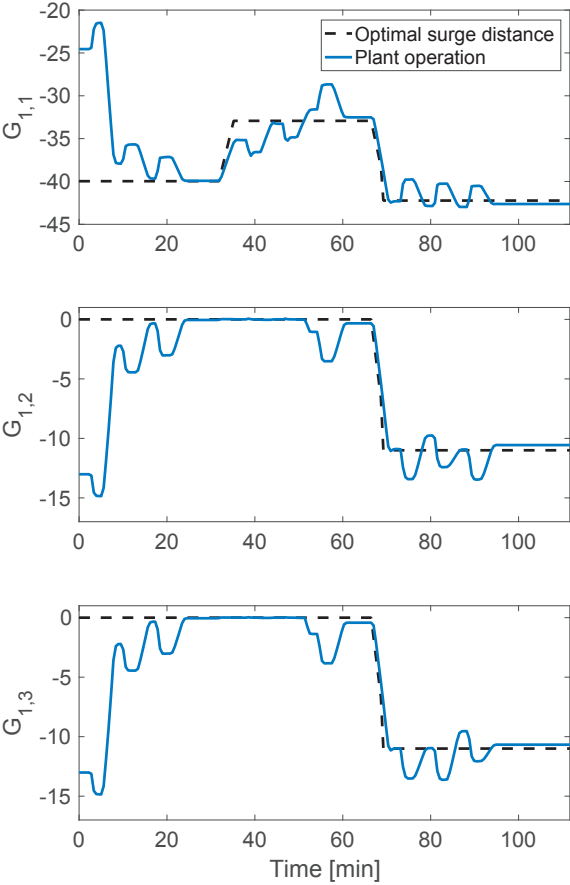


Figure 5.11 – Values of the plant surge control distance for serial configuration (blue line). Surge distance corresponding to the plant optimum (black dashed line).

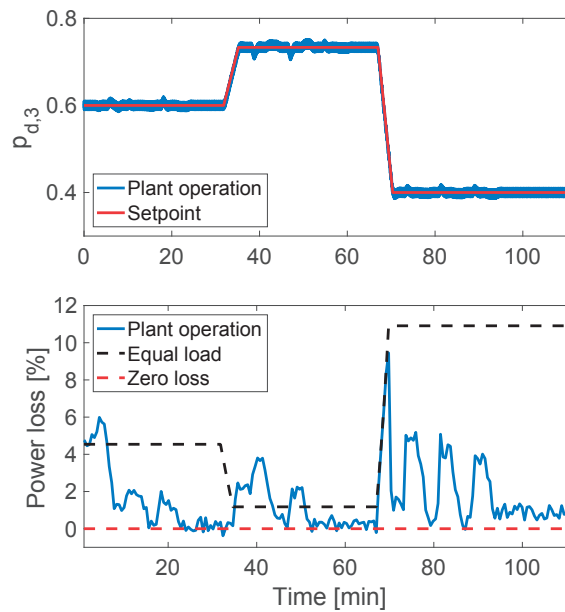


Figure 5.12 – **Top plot:** Normalized discharge pressure $p_{d,p,3}$. **Bottom plot:** Power loss using MA compared to equal-load distribution. The plant optimum corresponds to zero power loss.

5.5 Conclusion

This chapter has investigated the use of MA for load-sharing optimization in gas compression stations consisting of several compressors in both parallel and serial configurations. The analysis showed that *optimal operation of the plant* can be obtained after a few RTO iterations without having to update the model parameters or the compressor maps. Analysis showed that, by *using only local subsystem derivatives*, the algorithm is capable of quickly converging to the plant optimum. In fact, each subsystem relies on the estimation of the local power consumption, discharge pressure and mass flow derivatives with respect to its own inputs. What makes MA particularly well suited for this problem is that it is possible to excite all compressor speeds simultaneously without significantly perturbing the station setpoint. A solution to the load-sharing optimization problem was also proposed for the case where the compressors may operate on the surge control line. Furthermore, an efficient approach for obtaining accurate gradient estimates of the plant cost and constraints in the presence of noise and plant-model mismatch was discussed. The simulation case studies showed that the potential cost saving is up to 10% for the considered operating conditions.

6 Conclusions

6.1 Final Remarks

This thesis discussed difficulties in achieving plant optimality when the model used in optimization is inaccurate or in the presence of process disturbances. The thesis investigated the modifier-adaptation methodology for plants composed of multiple subsystems or units that are physically interconnected, such that the outputs of one system influence the inputs of other subsystems. Centralized RTO schemes may not be applicable or desirable for interconnected systems due to the complexity of the available model. Thus, we proposed a problem formulation for the overall plant relying on interconnection variables. Moreover, the thesis discussed reformulation and decomposition of the centralized RTO problem so that distributed optimization strategies can be applied. Since the available model is a simplified representation of the reality, the role of local measurements of the interconnection variables was discussed. Furthermore, the availability of local measurements and interconnection model led to the formulation of novel MA schemes using an alternative definition of modifiers. The main benefit of this formulation is that each subsystem requires modifiers only with respect to its own inputs. However, local plant gradients with respect to interconnection variables are needed. The problem is that interconnection variables are not directly manipulated, and careful steps had to be taken in the perturbation design. We showed that the proposed formulation using the interconnection model and local gradients could reduce the required number of setpoint changes for gradient estimation, thus implying faster convergence to optimality. In particular, the practical aspects and the effectiveness of the algorithms were illustrated via industry-inspired case study of load-sharing optimization. This case study showed that the algorithm is capable of quickly converging to plant optimality using only local modifier estimates.

The following paragraphs summarize the main conclusions of the chapters hereof:

Chapter 3: Modifier Adaptation for Interconnected Systems using Global Modifiers. This chapter has discussed the reformulation and the decomposition of the centralized optimization problem so that decentralized optimization strategies can be applied. It presented two

Chapter 6. Conclusions

distributed MA schemes for the real-time optimization of interconnected systems. For both schemes, plant measurements were utilized in place of an interconnection model. It was shown that, upon convergence, the computed inputs optimize the steady-state performance of the interconnected plant.

Both schemes demonstrated the strength of the modifier-adaptation methodology even when there is no model for the subsystem interconnections. The model that is used does not assume interaction between the subsystems. In turn, this leads to an increased number of RTO iterates. Furthermore, the proposed algorithms require the knowledge of the sensitivities of each subsystem's cost and constraint functions with respect to the inputs of the other subsystems. In practice, depending on the dimensionality of the system, the dynamics of the whole plant and the time it takes to reach steady state, obtaining these estimates can be very costly. Note that, the complexity of estimating these plant gradients is the same as in the case of treating the whole plant as one unit.

Chapter 4: Modifier Adaptation for Interconnected Systems using Local Modifiers. This chapter has extended the MA framework for interconnected systems, presented in Chapter 3, to a scheme that uses only local input measurements of each subsystem for plant gradient estimates. This approach allows the algorithm to overcome the challenges of most distributed algorithms in the context of RTO, in particular the dependence on global plant gradients and the availability of a global model. Two MA algorithms were discussed. Both MA schemes use *the interconnection model* as well as local plant gradient information with respect to *the decision variables* and *the interconnection variables*. We showed that, they deliver optimal steady-state performance for the overall plant. The first one is a centralized scheme, while the second scheme is a coordinator-based distributed MA scheme. The proposed distributed algorithm has a two-layer structure. In the model-based inner loop, a coordinator ensures that the inputs of the local subproblems are consistent with the interconnection model. Furthermore, feasible-side convergence of a distributed modifier scheme was shown, under the assumption of perfect plant gradient information.

Chapter 5: Application to Gas-Compressor Stations. This chapter has investigated the use of MA for load-sharing optimization in gas-compression stations consisting of several compressors in both parallel and serial configurations. The centralized version of MA for interconnected systems using local modifiers, presented in Chapter 4, is applied to both configurations. This analysis showed that, the algorithm is capable of quickly converging to the plant optimum by using only local subsystem derivatives. In fact, each subsystem relies on the estimation of the local power consumption, discharge pressure and mass flow derivatives with respect to its own inputs. A solution to the load-sharing optimization problem was proposed when the compressors may also operate on the surge control line. Also, an efficient approach for obtaining accurate gradient estimates of the plant cost and constraints in the presence of noise and plant-model mismatch was discussed. Furthermore, the complexity of this estimation is independent of the number of compressors. The MA algorithm works in synergy with a low-level controller that tracks the required discharge pressure in serial configuration

or the required mass flow in parallel configuration.

6.2 Outlook and Perspectives

Modifier-adaptation methods for centralized systems are well developed from a theoretical perspective. New challenges, inspired by practice and applications to industrial systems, will emerge and thus provide motivation for further improvements of the method. At this point, it is necessary to apply the methodology to experimental systems. Also, industrial practitioners will be more likely to adopt these methods if they have already been shown to work on real systems.

This thesis concludes with an outlook and suggestions.

Adaptation of outputs: There is one interesting observation that distinguishes the proposed methodology for interconnected systems using local gradients from other MA approaches, namely, the adaptation of equality constraints. Except for the recent work of Pappas et al. (2017), none of the methods in MA framework use the adaptation of equality constraints. These authors have investigated that the adaptation of outputs, with respect to global gradients, improves the convergence of an MA scheme. This analysis can be extended to the plants with interconnected systems and local gradient adaptation.

However, in this thesis, the utilization of modifiers for the adaptation of equality constraints is twofold. Firstly, the adaptation is required so that the modified model, using local modifiers, is capable of reaching a KKT point of the plant, as shown in Proposition 4.1. Also, this clearly affects the convergence of the outer-loop in distributed algorithms, as shown in Section 4.3. Secondly, the convergence of the inner-loop to the optimal value of dual variables is influenced by this adaptation. Namely, the update of the output model affects the gradients of the dual variables used in the model-based inner-loop optimization problem. This is something that has not been discussed in the ISOPE approaches (Brdyś and Tatjewski, 2005). If obtaining the solution to the inner-loop problem is computationally too intensive, even for utilization of more sophisticated distributed approaches, one may consider defining stopping criteria for the inner loop. This way, the inputs applied to the plant are not the optimal inputs of the current modified model. However, one must analyze whether this approach is robust enough to lead the plant towards its optimality. Similar problems have been tackled in the context of numerical optimization for fast inexact decomposition algorithms (Tran-Dinh et al., 2016; Pu et al., 2014).

Steady-state gradient estimation: As already mentioned, experimental gradient estimation is the main difficulty MA is facing. For processes with a large number of inputs, the challenge is to estimate many gradients from sparse and noisy data. By introducing local gradients, the curse of dimensionality can be reduced, but the estimates of the plant cost with respect to the interconnection variables are required. However, these variables are not directly manipulated. In this thesis, we analyzed the gradient estimation problem for a particular case of

compressor plants.

Nonetheless, an important question remains. *Is there a more general way to design an RTO scheme for efficient estimation of these gradients?* One way is to tackle this problem via Directional Modifier Adaptation (DMA) (Costello et al., 2016). In DMA, the gradients are corrected only in the subspace spanned by the most sensitive input directions to a small parametric perturbation. These directions are also called privileged directions. It is important to note that with the decomposition of optimization problem for interconnected systems, the privileged local input directions for each subsystem can be obtained from local parametric sensitivity analysis. These directions can be utilized in Dual Directional Modifier Adaptation, by introducing augmentation terms in each subproblem to reward steps in these privileged directions. Another approach would be to formulate the optimization problem with Dual MA (Marchetti et al., 2010; Rodger and Chachuat, 2010), where steps taken by an MA algorithm must be severely constrained to ensure good gradient estimates. This usually results in formulation of separate optimization problems that, in the case of interconnected systems, must be taken with precaution.

Another question that is also relevant for the herein proposed formulation is the following: *How to decide if estimates of local modifiers are less complicated to obtain than estimates of global gradients?* It is not clear how to give an answer as it depends on the particular case study, the topology of the network and the ability of an RTO scheme to provide sufficient excitation to the interconnected variables for accurate gradient estimates. Thus, this remains an open research direction.

Gradient estimation using transient measurements: MA techniques presented in this thesis are characterized by their ability to enforce plant optimality upon convergence despite the presence of model uncertainty. They are based on correcting the available model using gradient estimates computed from steady-state points at each iteration. With many iterations and inputs, this can make convergence to the plant optimum rather slow. Thus, it is of great interest to improve this approach by using a dynamic model as well as transient measurements for gradient computations. The idea is to implement steady-state MA in the transient phase, thereby attempting to reach optimality in a single transient operation to steady state (François and Bonvin, 2013; de Avila Ferreira et al., 2017). To that end, the steady-state optimization problem is solved repeatedly online, with the steady-state modifiers being estimated from transient measurements. The latter are used as if they were steady-state measurements. It is worth investigating whether local modifiers for the plant with interconnected systems can benefit from the use of local transient measurements. This way, the challenging gradient estimation with respect to interconnection variables can be avoided and, potentially, plant optimality can be reached in a single transient operation.

Inexact feasible-side distributed MA schemes: It is also worth noting that the role of second-order terms, introduced in the distributed MA Algorithm 4.3, is very similar to the the role of second-order terms presented in Appendix A for centralized MA with convex upper bounds.

In the case of large gradient error, this analysis can be extended to the adaptation of $\delta^{G_{i,j}}$ and δ^{Φ_i} in Proposition 4.3 (used in MA Algorithm 4.3) in order to improve the plant performance and reduce the input oscillations.

Subsystems with different dynamics: Distributed modifier adaption for interconnected systems using local modifiers may be very useful for tackling steady-state optimizing control of plants consisting of subsystems with different dynamics. Namely, complex plants usually consist of units whose settling times are different. The distributed methodology presented in Section 4.3 assumes synchronous optimization of all subsystems. This means that the coordinator has to wait for all units to reach steady state to update and distribute a new price vector (dual variables) to the subsystem optimizer. However, this should not always be the case as some units have smaller settling times in comparison to other units that are also part of the plant. The potential benefits of these methods would be the improved performance of the entire plant, faster convergence to plant optimality in comparison to the centralized approach and the overall system robustness when one of the control units fails or an information link fails. Also, the use of the above-mentioned local transient measurements may be beneficial for local gradient estimation. Asynchronous parallel methods in distributed optimization could have spurred the theoretical analysis for these RTO approaches. However, the motivation for asynchronous parallel methods in various fields of research differs from the one in RTO framework. Asynchronous approaches appear in the application area motivated by the existence of an inhomogeneous mixture of agents, where their local updates need not occur at a common rate (Stathopoulos and Jones, 2017). Also, these methods have been mostly motivated from memory allocation applications, when, e.g., a vector is stored in the shared memory space of a multicore computer and can be accessed and altered by the cores in an intermittent manner (Peng et al., 2016; Liu and Wright, 2015).

Finally, the general topic of real-time optimization of interconnected systems warrants additional investigation. This thesis has demonstrated theoretically that modifier adaptation can be applied to such systems. Thus, experimental validation is required as this will reveal where the challenges lie, and provide motivation for further improving the method.

A Modifier Adaptation with Convex Upper Bounds and Inexact Gradients

This appendix is based on:

P. Milosavljevic, R. Schneider, A. Cortinovis, T. Faulwasser, and D. Bonvin. A distributed feasible-side convergentmodifier-adaptation scheme for interconnected systems, with application to gas-compressor stations. *Comp. Chem. Eng.*, 115:474–486, 2018c.

Here, we recall the MA scheme with convex upper bounds presented in Subsection 2.2.2 and discuss its robustness in the case of inexact gradient information. Also, we analyze the convergence of this algorithm assuming plant cost gradient error. We finally demonstrate the effectiveness of the algorithm on a small numerical example and draw conclusions.

It is important to emphasize that much of the discussion in Section 2.2.2 has focused on very idealized cases, without considering how Algorithm 2.2 would behave in real applications, where accurate function and derivative values are not available. In the sequel, we will emphasize the robustness of Algorithm 2.2 to inexact gradients of the plant cost function. In static RTO, gradient estimation is rather expensive since it requires additional setpoint changes to the plant. For this analysis, we first introduce a function that expresses the mismatch between the plant and model costs.

The plant-model mismatch function $\Lambda(\mathbf{u}) : \mathbb{R}^{n_u} \mapsto \mathbb{R}$ is defined as

$$\Lambda(\mathbf{u}) = \phi_p(\mathbf{u}) - \phi(\mathbf{u}). \quad (\text{A.1})$$

Assumption A.1 (Differentiable model and plant functions).

Let the gradient of $\phi(\mathbf{u})$ and $\Lambda(\mathbf{u})$ be L^m - and L^Λ -Lipschitz continuous, respectively. Also, let $\phi(\mathbf{u})$ be a convex function. \square

Assumption A.2 (Cost gradient error).

The model gradient $\nabla\phi(\mathbf{u})$ is known so that the error in the gradient of $\Lambda(\mathbf{u})$ result only from error in the plant gradient $\nabla\phi_p(\mathbf{u})$. \square

Appendix A. Modifier Adaptation with Convex Upper Bounds and Inexact Gradients

Assumption A.2 implies

$$\begin{aligned}\widehat{\nabla\Lambda}(\mathbf{u}_k) &= \widehat{\nabla\phi_p}(\mathbf{u}_k) - \nabla\phi(\mathbf{u}_k) \\ &= \nabla\phi_p(\mathbf{u}_k) + \mathbf{e}_k - \nabla\phi(\mathbf{u}_k),\end{aligned}\tag{A.2}$$

where $\nabla\phi_p(\mathbf{u}_k)$ and $\widehat{\nabla\phi_p}(\mathbf{u}_k)$ are the exact and noisy gradients of the plant cost at the operating point \mathbf{u}_k , respectively, while $\nabla\phi(\mathbf{u}_k)$ is the exact gradient of the model cost, and \mathbf{e}_k is the plant cost gradient error.

The gradient of the Lagrangian function of Problem (2.9) at the optimal point \mathbf{u}_{k+1} is

$$\nabla\phi(\mathbf{u}_{k+1}) + \widehat{\nabla\Lambda}(\mathbf{u}_k) + \delta^\phi(\mathbf{u}_{k+1} - \mathbf{u}_k) + \mathbf{v}_k = \mathbf{0},\tag{A.3a}$$

where

$$\mathbf{v}_k = \sum_{j=1}^{n_g} (\mu_j^*)^\top \nabla g_{j,k}^U(\mathbf{u}_{k+1}),\tag{A.3b}$$

and μ_j^* is Lagrange multiplier associated with $g_{j,k}^U(\mathbf{u})$ at \mathbf{u}_{k+1} .

Proposition A.1 (Inexact gradient information). *Consider the modifier-adaptation scheme (2.9), the noisy gradient of the mismatch function $\tilde{\Lambda}(\mathbf{u})$ in eq. (A.2), and the vector \mathbf{v}_k defined in (A.3b). Let Assumptions 2.1, 2.2, A.1 and A.2 hold. It follows that:*

(i) *In the absence of gradient error ($\mathbf{e}_k = \mathbf{0}$), and if $\delta^\phi \geq L^\Lambda$, then the plant cost decreases at iteration $k + 1$.*

(ii) *If $\delta^\phi \geq L^\Lambda$, $\delta^\phi \geq L^m$ such that*

$$\|\widehat{\nabla\phi_p}(\mathbf{u}_k) + \mathbf{v}_k\|_2 \geq K(\delta^\phi) \|\mathbf{e}_k\|_2\tag{A.4}$$

where

$$K(\delta^\phi) = \frac{(\delta^\phi + L^m)^2}{(\delta^\phi - L^\Lambda)(\delta^\phi - L^m)},\tag{A.5}$$

then the plant cost decreases at iteration $k + 1$.

Proof. For the proof, we need to define the feasible sets associated with the upper-bounding constraint functions in (2.9) is given as

$$\mathcal{F}_{j,k}^U = \{\mathbf{u} \in \mathbb{R}^{n_u} : g_{j,k}^U(\mathbf{u}) \leq 0\}, \quad j = 1, \dots, n_g.\tag{A.6a}$$

Hence, the feasible convex set for the k -th RTO iteration can be written as

$$\mathcal{F}_k^U = \left(\bigcap_{j=1}^{n_g} \mathcal{F}_{j,k}^U \right). \quad (\text{A.6b})$$

We are now ready to prove Proposition A.1.

Part (i): Using that $\varepsilon_k^\phi = \Lambda(\mathbf{u}_k)$ and $\boldsymbol{\lambda}_k^\phi = \widehat{\nabla\Lambda}(\mathbf{u}_k)$ as well as the definition of optimality of (2.9) at \mathbf{u}_{k+1} , implies:

$$\begin{aligned} \phi(\mathbf{u}_{k+1}) + \Lambda(\mathbf{u}_k) + \widehat{\nabla\Lambda}(\mathbf{u}_k)^\top (\mathbf{u}_{k+1} - \mathbf{u}_k) + \frac{\delta^\phi}{2} \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2^2 \\ \leq \phi(\mathbf{u}) + \Lambda(\mathbf{u}_k) + \nabla\tilde{\Lambda}(\mathbf{u}_k)^\top (\mathbf{u} - \mathbf{u}_k) + \frac{\delta^\phi}{2} \|\mathbf{u} - \mathbf{u}_k\|_2^2, \end{aligned} \quad (\text{A.7a})$$

for all $\mathbf{u} \in \mathcal{F}_k^U$. Since we assume perfect constraint gradient information, the set \mathcal{F}_k^U is equivalent to the feasible set defined in (Marchetti et al., 2017, Thm.1). Hence, $\mathbf{u}_k \in \mathcal{F}_k^U$ and, for $\mathbf{u} = \mathbf{u}_k$, (A.7a) yields the inequality

$$\phi(\mathbf{u}_{k+1}) - \phi(\mathbf{u}_k) + \widehat{\nabla\Lambda}(\mathbf{u}_k)^\top (\mathbf{u}_{k+1} - \mathbf{u}_k) + \frac{\delta^\phi}{2} \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2^2 \leq 0. \quad (\text{A.7b})$$

Lipschitz continuity of the gradient of function $\Lambda(\mathbf{u})$ (Assumption A.1) implies:

$$\Lambda(\mathbf{u}_{k+1}) \leq \Lambda(\mathbf{u}_k) + \nabla\Lambda(\mathbf{u}_k)^\top (\mathbf{u}_{k+1} - \mathbf{u}_k) + \frac{L^\Lambda}{2} \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2^2.$$

By adding this last inequality to (A.7b) and rearranging the various terms gives:

$$\begin{aligned} \Lambda(\mathbf{u}_{k+1}) + \phi(\mathbf{u}_{k+1}) - \Lambda(\mathbf{u}_k) - \phi(\mathbf{u}_k) + (\widehat{\nabla\Lambda}(\mathbf{u}_k) - \nabla\Lambda(\mathbf{u}_k))^\top (\mathbf{u}_{k+1} - \mathbf{u}_k) \\ + \frac{(\delta^\phi - L^\Lambda)}{2} \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2^2 \leq 0. \end{aligned} \quad (\text{A.7c})$$

Recalling from (A.1) that $\Lambda = \phi_p - \phi$ and using the definition of the gradient error \mathbf{e} in (A.2), (A.7c) becomes

$$\phi_p(\mathbf{u}_{k+1}) - \phi_p(\mathbf{u}_k) + \mathbf{e}_k^\top (\mathbf{u}_{k+1} - \mathbf{u}_k) + \frac{(\delta^\phi - L^\Lambda)}{2} \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2^2 \leq 0. \quad (\text{A.7d})$$

In the case of perfect gradient estimation $\widehat{\nabla\Lambda}(\mathbf{u}_k) = \nabla\Lambda(\mathbf{u}_k)$, that is $\mathbf{e}_k = \mathbf{0}$, the following holds:

$$\phi_p(\mathbf{u}_{k+1}) - \phi_p(\mathbf{u}_k) + \frac{(\delta^\phi - L^\Lambda)}{2} \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2^2 \leq 0.$$

Hence, for $\delta^\phi - L^\Lambda \geq 0$, we obtain:

$$\phi_p(\mathbf{u}_{k+1}) \leq \phi_p(\mathbf{u}_k).$$

Appendix A. Modifier Adaptation with Convex Upper Bounds and Inexact Gradients

Part (ii): Writing down the optimality conditions for Problem (2.9) at the point \mathbf{u}_{k+1} gives:

$$\nabla\phi(\mathbf{u}_{k+1}) + \widehat{\nabla\Lambda}(\mathbf{u}_k) + \delta^\phi(\mathbf{u}_{k+1} - \mathbf{u}_k) + \mathbf{v}_k = \mathbf{0}. \quad (\text{A.8a})$$

with the vector \mathbf{v}_k defined as in (A.3b). Based on this equation, we will now derive upper and lower bounds on the term $\|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2$. For the upper bound, we substitute $\widehat{\nabla\Lambda}(\mathbf{u}_k)$ by the expression $\widehat{\nabla\Lambda}(\mathbf{u}_k) = \widehat{\nabla\phi_p}(\mathbf{u}_k) - \nabla\phi(\mathbf{u}_k)$.

$$\mathbf{u}_{k+1} - \mathbf{u}_k = \frac{1}{\delta^\phi} \left(-\nabla\phi(\mathbf{u}_{k+1}) + \nabla\phi(\mathbf{u}_k) - \widehat{\nabla\phi_p}(\mathbf{u}_k) + \mathbf{v}_k \right). \quad (\text{A.8b})$$

Using the triangle inequality and Lipschitz continuity of $\nabla\phi(\mathbf{u})$ (Assumption A.1) gives:

$$\begin{aligned} \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2 &\leq \frac{1}{\delta^\phi} \|\nabla\phi(\mathbf{u}_{k+1}) - \nabla\phi(\mathbf{u}_k)\|_2 + \frac{1}{\delta^\phi} \|\widehat{\nabla\phi_p}(\mathbf{u}_k) + \mathbf{v}_k\|_2 \\ &\leq \frac{L^m}{\delta^\phi} \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2 + \frac{1}{\delta^\phi} \|\widehat{\nabla\phi_p}(\mathbf{u}_k) + \mathbf{v}_k\|_2. \end{aligned} \quad (\text{A.8c})$$

Hence, for $\delta^\phi > L^m$, we get the desired upper bound

$$\|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2 \leq \frac{1}{(\delta^\phi - L^m)} \|\widehat{\nabla\phi_p}(\mathbf{u}_k) + \mathbf{v}_k\|_2. \quad (\text{A.8d})$$

In order to derive the lower bound, we first rewrite the equality (A.8b) as

$$\delta^\phi(\mathbf{u}_{k+1} - \mathbf{u}_k) + \nabla\phi(\mathbf{u}_{k+1}) - \nabla\phi(\mathbf{u}_k) = -\widehat{\nabla\phi_p}(\mathbf{u}_k) + \mathbf{v}_k \quad (\text{A.9a})$$

Taking the norm on both sides, and using Lipschitz continuity and the triangle inequality gives:

$$\delta^\phi \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2 + L^m \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2 \geq \quad (\text{A.9b})$$

$$\delta^\phi \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2 + \|\nabla\phi(\mathbf{u}_{k+1}) - \nabla\phi(\mathbf{u}_k)\|_2 \geq \quad (\text{A.9c})$$

$$\|\widehat{\nabla\phi_p}(\mathbf{u}_k) + \mathbf{v}_k\|_2. \quad (\text{A.9d})$$

Simplifying (A.9b), we obtain the desired lower bound

$$\|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2 \geq \frac{1}{(\delta^\phi + L^m)} \|\widehat{\nabla\phi_p}(\mathbf{u}_k) + \mathbf{v}_k\|_2. \quad (\text{A.9e})$$

Going back to (A.7d), we have

$$\phi_p(\mathbf{u}_{k+1}) - \phi_p(\mathbf{u}_k) + (\mathbf{e}_k)^\top (\mathbf{u}_{k+1} - \mathbf{u}_k) + \frac{(\delta^\phi - L^\Lambda)}{2} \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2^2 \leq 0. \quad (\text{A.10a})$$

Rearranging we get:

$$\phi_p(\mathbf{u}_{k+1}) - \phi_p(\mathbf{u}_k) \leq -(\mathbf{e}_k)^\top (\mathbf{u}_{k+1} - \mathbf{u}_k) - \frac{(\delta^\phi - L^\Lambda)}{2} \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2^2 \quad (\text{A.10b})$$

$$\leq \|\mathbf{e}_k\|_2 \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2 - \frac{(\delta^\phi - L^\Lambda)}{2} \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2^2. \quad (\text{A.10c})$$

Next, (a) we multiply (A.8d) by $\|\mathbf{e}_k\|_2$, and (b) we square (A.9e) and multiply the result by $-\frac{(\delta^\phi - L^\Lambda)}{2}$. Then, we use both (a) and (b) in (A.10c) to obtain:

$$\phi_p(\mathbf{u}_{k+1}) - \phi_p(\mathbf{u}_k) \leq \frac{1}{\delta^\phi - L^m} \|\mathbf{e}_k\|_2 \|\widehat{\nabla\phi}_p(\mathbf{u}_k) + \mathbf{v}_k\|_2 - \frac{\delta^\phi - L^\Lambda}{(\delta^\phi + L^m)^2} \|\widehat{\nabla\phi}_p(\mathbf{u}_k) + \mathbf{v}_k\|_2^2 \quad (\text{A.10d})$$

$$\leq \|\widehat{\nabla\phi}_p(\mathbf{u}_k) + \mathbf{v}_k\|_2 \left(\frac{1}{\delta^\phi - L^m} \|\mathbf{e}_k\|_2 - \frac{\delta^\phi - L^\Lambda}{(\delta^\phi + L^m)^2} \|\widehat{\nabla\phi}_p(\mathbf{u}_k) + \mathbf{v}_k\|_2 \right). \quad (\text{A.10e})$$

The right-hand side of (A.10e) is negative for $\|\widehat{\nabla\phi}_p(\mathbf{u}_k) + \mathbf{v}_k\|_2 \geq K \|\mathbf{e}_k\|_2$, where

$$K = \frac{(\delta^\phi + L^m)^2}{(\delta^\phi - L^\Lambda)(\delta^\phi - L^m)}.$$

Then, for $\|\widehat{\nabla\phi}_p(\mathbf{u}_k) + \mathbf{v}_k\|_2 \geq K \|\mathbf{e}_k\|_2$, the plant cost function decreases at iteration $k + 1$. \square

Remark A.1 (Plant cost decrease). *The result of Proposition A.1 indicates that the plant cost will decrease until the input converges to the region where*

$$\|\widehat{\nabla\phi}_p(\mathbf{u}_k) + \mathbf{v}_k\|_2 < K (\delta^\phi) \|\mathbf{e}_k\|_2. \quad (\text{A.11})$$

In this region, the algorithm can behave quite unpredictably due to the presence of random gradient errors. If the errors vary substantially, the method will tend to oscillate within the region in which the plant optimum lies. According to (A.3), in case the optimum \mathbf{u}_{k+1} lies on the constraints, \mathbf{v}_k in (A.3b) is a non-zero vector that can be obtained from the model optimization problem (2.9). Hence, the proof indicates that, in the case of an unconstrained optimum, the condition in (A.4) reduces to $\|\widehat{\nabla\phi}_p(\mathbf{u}_k)\|_2 \geq K (\delta^\phi) \|\mathbf{e}_k\|_2$. If exact value of gradient error norm $\|\mathbf{e}_k\|_2$ is not known, one can be obtained its bound as in Subsection 2.2.4.

Next, we discuss the influence of the parameter δ^ϕ on the performance of the MA Algorithm 2.2. Note that the choice of δ^ϕ is bounded from below by the parameters L^Λ and L^m . The estimation of L^Λ is very challenging in practice; in general, it is even more difficult than obtaining exact plant gradients. On the one hand, imposing monotonic decrease of the plant cost would require conservative values of δ^ϕ , especially if the gradients are inexact and if the noise level is high. On the other hand, choosing high values of δ^ϕ results in slow convergence of Algorithm 2.2, thus requiring too many setpoint changes on the real system.

We will now show how Condition (A.4) can help in reduce the number of setpoint changes.

Appendix A. Modifier Adaptation with Convex Upper Bounds and Inexact Gradients

Namely, we will propose modified version of MA Algorithm 2.2 that adapts the parameter δ^ϕ in order to speedup convergence to the neighborhood of the plant optimum. Here, we choose large values of δ^ϕ , namely to give certain robustness to the MA algorithm in presence of large gradient error, thus preventing large steps and implying slow convergence. However, if Condition (A.4) holds, we can adapt the value of δ^ϕ such that the inequality (A.4) holds with equality. In this case, we compute δ^ϕ from

$$\|\widehat{\nabla\phi_p}(\mathbf{u}_k) + \mathbf{v}_k\|_2 = K(\delta^\phi) \|\mathbf{e}_k\|_2 \quad (\text{A.12})$$

and utilize it in the MA Algorithm 2.2. This way, the MA scheme will take larger steps knowing that the plant cost decrease Condition (A.4) still holds. Note that $K(\delta^\phi)$ in (A.5) is a decreasing function of δ^ϕ for $\delta^\phi \geq L^\Lambda$, $\delta^\phi \geq L^m$ and the solution of (A.12) is unique.

Hence, we propose the following Algorithm A.1 as the modified version of the MA Algorithm 2.2 in Section 2.2.2

Algorithm A.1 : MA Algorithm 2.2 with adaptation of δ^ϕ

1. *Initialization*: Provide the initial point \mathbf{u}_0 and $\delta_{max}^\phi, L^\Lambda$ and L^m . Set $k := 0$.
 2. *Plant experiment*: Apply the set of inputs \mathbf{u}_k to the plant and wait for steady state.
 3. *Modifier computation*: Compute the modifiers as per eqs. (2.5a)–(2.5c).
 4. *Adaptation of δ^ϕ* :
 Set $\delta^\phi = \delta_{max}^\phi$
 if eq. (A.4) holds
 Compute new δ^ϕ from eq. (A.12)
 end
 5. *New input calculation*: Compute \mathbf{u}_{k+1} by solving Problem (2.9).
 6. *Iterate*: Set $k := k + 1$ and return to Step 2.
-

A.1 Numerical Example

To illustrate the influence of the parameter δ^ϕ on the convergence of the MA scheme, we present a simple example characterized by an unconstrained optimum.

Consider the plant optimization problem:

$$\min_{u_1, u_2} \phi_p := 1.8(u_1 + 1)^2 + 2(u_2 + 1)^2 \quad (\text{A.13a})$$

$$\text{s.t. } -5 \leq u_1 \leq 5 \quad (\text{A.13b})$$

$$-5 \leq u_2 \leq 5, \quad (\text{A.13c})$$

and its model counterpart:

$$\min_{u_1, u_2} \phi := 2.5(u_1 - 0.5)^2 + 1.3(u_2 - 0.5)^2 \tag{A.14a}$$

$$\text{s.t. } -5 \leq u_1 \leq 5 \tag{A.14b}$$

$$-5 \leq u_2 \leq 5. \tag{A.14c}$$

We discuss various scenarios with different δ^ϕ values as presented in Table A.1 and Figure A.1. Each scenario considers a constant norm of the gradient error \mathbf{e} .

Scenario 1 illustrates the case where the small value $\delta_{max}^\phi = 3$ is chosen. Algorithm 2.2 with no adaptation of δ^ϕ is considered. Here $\delta^\phi = 3$ is a value that ensures monotonic cost decrease in the absence of gradient error ($\mathbf{e} = \mathbf{0}$). However, once in the vicinity of the optimum, after about 10 RTO iterations, the algorithm starts oscillating significantly due to the large gradient error $\|\mathbf{e}\| = 6$. For the same value $\delta_{max}^\phi = 3$, applying Algorithm A.1 will not change the outcome of Scenario 1. This is due to the fact that, for $\|\mathbf{e}\| = 6$ and the small value $\delta_{max}^\phi = 3$, Condition (A.4) is never satisfied.

Scenarios 2 and 3 correspond to RTO iterates δ^ϕ being adapted. The same value of gradient error $\|\mathbf{e}_k\| = 6$, as in Scenario 1, is considered. In Figure A.1, the dotted line represents the value of $K(\delta_{max}^\phi)\|\mathbf{e}\|_2$. Thus, whenever $\|\widehat{\nabla\phi}_p(\mathbf{u}_k)\|_2 \geq K(\delta_{max}^\phi)\|\mathbf{e}_k\|_2$ holds, the value of δ^ϕ is adapted according to Algorithm A.1. Due to this adaptation, the vicinity of the optimum is reached after about 15-20 iterations. However, there are no oscillations due to gradient error.

In Scenario 4, since δ^ϕ is not adapted, the convergence to the neighborhood of the optimum is slower in comparison to Scenarios 2 and 3.

	$K(\delta_{max}^\phi)\ \mathbf{e}\ $	δ_{max}^ϕ	Algorithm	$\ \mathbf{e}\ $
Scenario 1	181.5	3	Alg. 2.2	6
Scenario 2	7.1	50	Alg. A.1	6
Scenario 3	8	30	Alg. A.1	6
Scenario 4	7.1	50	Alg. 2.2	6

Table A.1 – Four different scenarios corresponding to Figure A.1

Appendix A. Modifier Adaptation with Convex Upper Bounds and Inexact Gradients

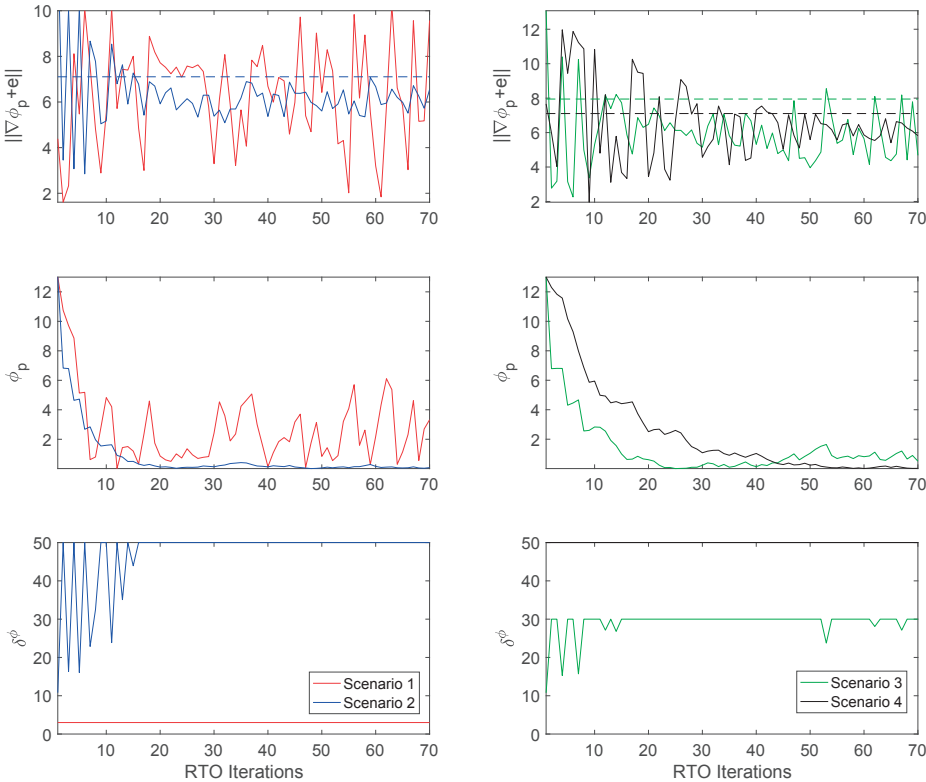


Figure A.1 – RTO iterates for Algorithm 2.2 and scenarios in Table A.1

This appendix has discussed the robustness of an MA algorithm with convex upper bounds in the case of inexact gradient information. We have analyzed the convergence of this algorithm assuming gradient error in the plant cost. The conditions that can be used to reduce the conservatism of the proposed algorithm have been derived. This way, convergence to the neighborhood of the plant optimum can be reached faster.

B A Link Between MA and Proximal-Gradient Algorithm

This appendix recalls the basic proximal-gradient algorithm, which has been shown to be a very powerful tool for solving nonconvex and nonsmooth problems that occur in many applications. For the purpose of this paper, we briefly recall its application for minimizing the sum of two functions (Bolte et al., 2014; Hours and Jones, 2016).

Consider the following optimization problem:

$$\min_{\mathbf{u} \in \Omega} t(\mathbf{u}) + h(\mathbf{u}), \quad (\text{B.1})$$

where Ω is a convex set in \mathbb{R}^{n_u} , $t(\mathbf{u}) : \mathbb{R}^{n_u} \rightarrow \mathbb{R}$ is a convex function, not necessarily smooth, and $h(\mathbf{u}) : \mathbb{R}^{n_u} \rightarrow \mathbb{R}$ is a differentiable function. To move the input constraints to the objective function, we will replace $t(\mathbf{u})$ by the penalty function

$$f(\mathbf{u}) := t(\mathbf{u}) + \mathcal{I}_{\Omega}(\mathbf{u}). \quad (\text{B.2})$$

where $\mathcal{I}_{\Omega}(\mathbf{u})$ is the indicator function of convex set Ω . As a result, we may rewrite (B.1) as the unconstrained problem:

$$\min_{\mathbf{u} \in \mathbb{R}^{n_u}} f(\mathbf{u}) + h(\mathbf{u}). \quad (\text{B.3})$$

Definition B.1 (Proximal operator). *Let $f : \mathbb{R}^{n_u} \rightarrow \mathbb{R}$ be a lower-semicontinuous convex function and $L > 0$. The proximal operator associated with the function $f(\mathbf{u})$ and the coefficient L , denoted prox_L^f , is defined as follows:*

$$\text{prox}_L^f(c) = \underset{\mathbf{u} \in \mathbb{R}^{n_u}}{\text{argmin}} \quad f(\mathbf{u}) + \frac{L}{2} \|\mathbf{u} - c\|_2^2. \quad (\text{B.4})$$

Then, the solution to Problem (B.3) can be obtained via the proximal-gradient algorithm (PGA) (Parikh and Boyd, 2014):

We shall now investigate the relationship between the modifier-adaptation Algorithm 2.2 and

Appendix B. A Link Between MA and Proximal-Gradient Algorithm

Algorithm B.1 : Proximal-gradient algorithm

1. *Initialization:* Provide the initial point \mathbf{u}_0 . Set $k := 0$.
2. *New point calculation:*

$$\mathbf{u}_{k+1} = \text{prox}_L^f \left(\mathbf{u}_k - \frac{1}{L} \nabla h(\mathbf{u}_k) \right). \quad (\text{B.5})$$

3. *Iterate:* Set $k := k + 1$ and return to Step 2.
-

the proximal-gradient Algorithm B.1. For this analysis, we first introduce a function that expresses the mismatch between the plant and model costs.

Note that the plant-model mismatch function $\Lambda(\mathbf{u})$ is defined in (A.1) and the feasible set associated with Problem (2.9) is given in (A.6).

The relationship between the MA Algorithm 2.2 and the proximal-gradient Algorithm B.1 can be stated as follows.

Proposition B.1 (Equivalence between MA and PGA).

Let Assumption A.1 hold. Consider the modifier-adaptation recursion (2.9) at the k -th iteration and the plant-model mismatch function $\Lambda(\mathbf{u})$ defined in (A.1). Then, if $L = \delta^\phi$, the modifier-adaptation Problem (2.9) is equivalent to the proximal-gradient update

$$\mathbf{u}_{k+1} = \text{prox}_L^\Phi \left(\mathbf{u}_k - \frac{1}{L} \nabla \Lambda(\mathbf{u}_k) \right) \quad (\text{B.6})$$

associated with $\Phi(\mathbf{u}) := \phi(\mathbf{u}) + \mathcal{I}_{\mathcal{F}_k^U}(\mathbf{u})$, where \mathcal{F}_k^U is the convex set defined in (A.6).

Proof. From the definition of the proximal operator (B.4):

$$\begin{aligned} \mathbf{u}_{k+1} &= \text{prox}_L^\Phi \left(\mathbf{u}_k - \frac{1}{L} \nabla \Lambda(\mathbf{u}_k) \right) \\ &= \underset{\mathbf{u} \in \mathcal{F}_k^U}{\text{argmin}} \quad \phi(\mathbf{u}) + \frac{L}{2} \|\mathbf{u} - \mathbf{u}_k + \frac{1}{L} \nabla \Lambda(\mathbf{u}_k)\|_2^2 \\ &= \underset{\mathbf{u} \in \mathcal{F}_k^U}{\text{argmin}} \quad \phi(\mathbf{u}) + \nabla \Lambda(\mathbf{u}_k)^\top (\mathbf{u} - \mathbf{u}_k) + \frac{L}{2} \|\mathbf{u} - \mathbf{u}_k\|_2^2 \\ &= \underset{\mathbf{u} \in \mathcal{F}_k^U}{\text{argmin}} \quad \phi(\mathbf{u}) + \Lambda(\mathbf{u}_k) + \nabla \Lambda(\mathbf{u}_k)^\top (\mathbf{u} - \mathbf{u}_k) + \frac{L}{2} \|\mathbf{u} - \mathbf{u}_k\|_2^2 \end{aligned}$$

Hence, \mathbf{u}_{k+1} minimizes $\phi(\mathbf{u})$ plus a quadratic local model of $\Lambda(\mathbf{u}_k)$ around \mathbf{u}_k . Since $\Lambda(\mathbf{u}_k) = \phi_p(\mathbf{u}_k) - \phi(\mathbf{u}_k) = \varepsilon_k^\phi$ and $\nabla \Lambda(\mathbf{u}_k) = \nabla \phi_p(\mathbf{u}_k) - \nabla \phi(\mathbf{u}_k) = \boldsymbol{\lambda}_k^\phi$, Problem (2.9) is equivalent to Problem (B.6) for $L = \delta^\phi$. \square

C Proofs

C.1 Proof of Lemma 3.1

Proof. Using Assumption 3.1, the plant optimization problem (3.1) can be equivalently rewritten as

$$(\mathbf{u}_p^*, \mathbf{v}_p^*, \mathbf{y}_p^*) = \underset{\mathbf{u}, \mathbf{v}_p, \mathbf{y}_p}{\operatorname{argmin}} \Phi_p(\mathbf{u}, \mathbf{v}_p, \mathbf{y}_p) \quad (\text{C.1a})$$

$$\text{s.t. } \mathbf{y}_p = \mathbf{f}_p(\mathbf{u}), \quad (\text{C.1b})$$

$$\mathbf{G}_p(\mathbf{u}, \mathbf{v}_p, \mathbf{y}_p) \leq \mathbf{0}, \quad (\text{C.1c})$$

$$\mathbf{v}_p = \mathbf{h}_p(\mathbf{u}). \quad (\text{C.1d})$$

Substituting the equality constraints (C.1b) and (C.1d) into the objective function (C.1a) and into the inequality constraints (C.1c), Problem (C.1) becomes

$$(\mathbf{u}_p^*, \mathbf{v}_p^*, \mathbf{y}_p^*) = \underset{\mathbf{u}, \mathbf{v}_p, \mathbf{y}_p}{\operatorname{argmin}} \Phi_p(\mathbf{u}, \mathbf{h}_p(\mathbf{u}), \mathbf{f}_p(\mathbf{u})) \quad (\text{C.2a})$$

$$\text{s.t. } \mathbf{y}_p = \mathbf{f}_p(\mathbf{u}), \quad (\text{C.2b})$$

$$\mathbf{G}_p(\mathbf{u}, \mathbf{h}_p(\mathbf{u}), \mathbf{f}_p(\mathbf{u})) \leq \mathbf{0}, \quad (\text{C.2c})$$

$$\mathbf{v}_p = \mathbf{h}_p(\mathbf{u}). \quad (\text{C.2d})$$

In this formulation, the optimal value of \mathbf{u} is no longer influenced by the variables \mathbf{y}_p and \mathbf{v}_p . Moreover, the optimal values \mathbf{y}_p^* and \mathbf{v}_p^* are now uniquely defined by the equality constraints (C.2b) and (C.2d). Hence, Problem (C.2) can be equivalently rewritten as

$$\mathbf{u}_p^* = \underset{\mathbf{u}}{\operatorname{argmin}} \Phi_p(\mathbf{u}, \mathbf{h}_p(\mathbf{u}), \mathbf{f}_p(\mathbf{u})) \quad (\text{C.3a})$$

$$\text{s.t. } \mathbf{G}_p(\mathbf{u}, \mathbf{h}_p(\mathbf{u}), \mathbf{f}_p(\mathbf{u})) \leq \mathbf{0}, \quad (\text{C.3b})$$

$$\mathbf{y}_p^* = \mathbf{f}_p(\mathbf{u}_p^*), \quad (\text{C.3c})$$

$$\mathbf{v}_p^* = \mathbf{h}_p(\mathbf{u}_p^*), \quad (\text{C.3d})$$

and finally as (3.4). \square

C.2 Proof of Proposition 3.1

Proof. Problem (3.11) can be rewritten as

$$\left(\mathbf{u}_{k+1}^*, \mathbf{v}_{k+1}^*, \mathbf{y}_{k+1}^*\right) = \underset{\mathbf{u}, \mathbf{v}, \mathbf{y}}{\operatorname{argmin}} \Phi(\mathbf{u}, \mathbf{v}, \mathbf{y}) + (\boldsymbol{\lambda}_k^\phi)^\top (\mathbf{u} - \mathbf{u}_k) \quad (\text{C.4a})$$

$$\text{s.t.} \quad \mathbf{y} = \mathbf{F}(\mathbf{u}, \mathbf{v}), \quad (\text{C.4b})$$

$$\mathbf{G}(\mathbf{u}, \mathbf{v}, \mathbf{y}) + \boldsymbol{\epsilon}_k^g + (\boldsymbol{\lambda}_k^g)^\top (\mathbf{u} - \mathbf{u}_k) \leq \mathbf{0}, \quad (\text{C.4c})$$

$$\mathbf{v} = \mathbf{H}(\mathbf{y}), \quad (\text{C.4d})$$

with the filter (3.7g). Applying Assumption 3.2 and using the definition of the functions $\phi(\cdot)$ and $\mathbf{g}(\cdot)$, the equivalent problem is obtained

$$\left(\mathbf{u}_{k+1}^*, \mathbf{v}_{k+1}^*, \mathbf{y}_{k+1}^*\right) = \underset{\mathbf{u}, \mathbf{v}, \mathbf{y}}{\operatorname{argmin}} \phi(\mathbf{u}) + (\boldsymbol{\lambda}_k^\phi)^\top (\mathbf{u} - \mathbf{u}_k) \quad (\text{C.5a})$$

$$\text{s.t.} \quad \mathbf{y} = \mathbf{f}(\mathbf{u}), \quad (\text{C.5b})$$

$$\mathbf{g}(\mathbf{u}) + \boldsymbol{\epsilon}_k^g + (\boldsymbol{\lambda}_k^g)^\top (\mathbf{u} - \mathbf{u}_k) \leq \mathbf{0}, \quad (\text{C.5c})$$

$$\mathbf{v} = \mathbf{h}(\mathbf{u}), \quad (\text{C.5d})$$

with the filter (3.7g). Similar to (C.2), the optimal value of \mathbf{u} does not depend on \mathbf{y} or \mathbf{v} . Hence, we can simplify further to obtain the equations

$$\mathbf{u}_{k+1}^* = \underset{\mathbf{u}}{\operatorname{argmin}} \phi(\mathbf{u}) + (\boldsymbol{\lambda}_k^\phi)^\top (\mathbf{u} - \mathbf{u}_k), \quad (\text{C.6a})$$

$$\text{s.t.} \quad \mathbf{g}(\mathbf{u}) + \boldsymbol{\epsilon}_k^g + (\boldsymbol{\lambda}_k^g)^\top (\mathbf{u} - \mathbf{u}_k) \leq \mathbf{0}, \quad (\text{C.6b})$$

$$\mathbf{y}_{k+1}^* = \mathbf{f}(\mathbf{u}_{k+1}^*), \quad (\text{C.6c})$$

$$\mathbf{v}_{k+1}^* = \mathbf{h}(\mathbf{u}_{k+1}^*), \quad (\text{C.6d})$$

with the filter (3.7g). Since (C.6a) and (C.6b) as well as the filter equations are the same as in the standard MA scheme (2.4), Theorem 2.2 applies and gives the desired result. \square

C.3 Proof of Proposition 3.2

Proof. Problem (3.17) can be rewritten as

$$\left(\mathbf{u}_{k+1}^*, \mathbf{y}_{k+1}^*\right) = \underset{\mathbf{u}, \mathbf{y}}{\operatorname{argmin}} \Phi(\mathbf{u}, \mathbf{v}_{p,k}, \mathbf{y}) + (\boldsymbol{\lambda}_k^\phi)^\top (\mathbf{u} - \mathbf{u}_k), \quad (\text{C.7a})$$

$$\text{s.t.} \quad \mathbf{y} = \mathbf{F}(\mathbf{u}, \mathbf{v}_{p,k}), \quad (\text{C.7b})$$

$$\mathbf{G}(\mathbf{u}, \mathbf{v}_{p,k}) + \boldsymbol{\epsilon}_k^g + (\boldsymbol{\lambda}_k^g)^\top (\mathbf{u} - \mathbf{u}_k) \leq \mathbf{0}, \quad (\text{C.7c})$$

with the filter (3.7g). Substituting $\mathbf{F}(\mathbf{u}, \mathbf{v}_{p,k})$ for \mathbf{y} in the objective function and inequality constraints and using the definition of the functions $\phi(\cdot)$ and $\mathbf{g}(\cdot)$, the equivalent problem is obtained

$$\left(\mathbf{u}_{k+1}^*, \mathbf{y}_{k+1}^* \right) = \underset{\mathbf{u}, \mathbf{y}}{\operatorname{argmin}} \phi(\mathbf{u}) + (\boldsymbol{\lambda}_k^\phi)^\top (\mathbf{u} - \mathbf{u}_k) \quad (\text{C.7d})$$

$$\text{s.t.} \quad \mathbf{y} = \mathbf{F}(\mathbf{u}, \mathbf{v}_{p,k}), \quad (\text{C.7e})$$

$$\mathbf{g}(\mathbf{u}) + \boldsymbol{\epsilon}_k^g + (\boldsymbol{\lambda}_k^g)^\top (\mathbf{u} - \mathbf{u}_k) \leq \mathbf{0}, \quad (\text{C.7f})$$

with the filter (3.7g). Again, the optimal value of \mathbf{u} does not depend on \mathbf{y} . Hence, we can simplify further to obtain the equations

$$\mathbf{u}_{k+1}^* = \underset{\mathbf{u}}{\operatorname{argmin}} \phi(\mathbf{u}) + (\boldsymbol{\lambda}_k^\phi)^\top (\mathbf{u} - \mathbf{u}_k), \quad (\text{C.7g})$$

$$\text{s.t.} \quad \mathbf{g}(\mathbf{u}) + \boldsymbol{\epsilon}_k^g + (\boldsymbol{\lambda}_k^g)^\top (\mathbf{u} - \mathbf{u}_k) \leq \mathbf{0}, \quad (\text{C.7h})$$

$$\mathbf{y}_{k+1}^* = \mathbf{F}(\mathbf{u}_{k+1}^*, \mathbf{v}_{p,k}), \quad (\text{C.7i})$$

with the filter (3.7g). Since (C.7g) and (C.7h) as well as the filter equations are the same as in the standard MA scheme (2.4), Theorem 2.2 applies and gives the desired result. \square

C.4 Proof of Proposition 4.1

Proof. The proof proceeds in two steps: the first part of the proof shows that, upon convergence, \mathbf{v}_∞ is equal to $\mathbf{v}_{p,\infty}$; second we show that a KKT point of the plant is reached.

Step 1: The following analysis shows that, upon convergence, the pair $(\mathbf{u}_\infty, \mathbf{v}_{p,\infty})$ is a solution to (4.5d). In fact, by evaluating equation (4.5d)

$$\mathbf{Q}_m(\mathbf{u}, \mathbf{v}) := \mathbf{v} - \left(\mathbf{H}(\mathbf{y}) + \boldsymbol{\epsilon}_k^H + (\boldsymbol{\lambda}_k^H)^\top (\mathbf{z} - \mathbf{z}_{p,k}) \right) \quad (\text{C.8})$$

at point $\mathbf{z}_{p,\infty} = [\mathbf{u}_\infty, \mathbf{v}_{p,\infty}]$, we obtain

$$\mathbf{Q}_m(\mathbf{u}_\infty, \mathbf{v}_{p,\infty}) = \mathbf{v}_{p,\infty} - \mathbf{H}(\mathbf{y}(\mathbf{u}_\infty, \mathbf{v}_{p,\infty})) - \boldsymbol{\epsilon}_k^H - (\boldsymbol{\lambda}_k^H)^\top (\mathbf{z}_{p,\infty} - \mathbf{z}_{p,\infty}) \quad (\text{C.9a})$$

$$= \mathbf{v}_{p,\infty} - \mathbf{H}_p(\mathbf{y}_p(\mathbf{u}_\infty, \mathbf{v}_{p,\infty})) \quad (\text{C.9b})$$

$$= \mathbf{0}. \quad (\text{C.9c})$$

Thus, from (C.9), the pair $(\mathbf{u}_\infty, \mathbf{v}_{p,\infty})$ is solution to (4.5d). Since the solution to (4.5d) is unique (Assumption 4.2), this means that upon convergence to \mathbf{u}_∞ , the duple $(\mathbf{u}_\infty, \mathbf{v}_\infty)$ is the same as $(\mathbf{u}_\infty, \mathbf{v}_{p,\infty})$.

Step 2: The second part of the proof is focused on the KKT conditions of problems (4.1)

and (4.5). The definition of the modifiers (4.5) implies that

$$\mathbf{G}_m = \mathbf{G}(\mathbf{u}_\infty, \mathbf{v}_{p,\infty}) + \boldsymbol{\varepsilon}_\infty^G = \mathbf{G}_p(\mathbf{u}_\infty, \mathbf{v}_{p,\infty}) \quad (\text{C.10a})$$

$$\begin{aligned} \nabla_{\mathbf{z}} \mathbf{G}_m(\mathbf{u}_\infty, \mathbf{v}_\infty) &= \nabla_{\mathbf{z}} \mathbf{G}_m(\mathbf{u}_\infty, \mathbf{v}_{p,\infty}) \\ &= \nabla_{\mathbf{z}} \mathbf{G}(\mathbf{u}_\infty, \mathbf{v}_{p,\infty}) + \boldsymbol{\lambda}_\infty^G \\ &= \nabla_{\mathbf{z}_p} \mathbf{G}_p(\mathbf{u}_\infty, \mathbf{v}_{p,\infty}) \end{aligned} \quad (\text{C.10b})$$

$$\begin{aligned} \nabla_{\mathbf{z}} \Phi_m(\mathbf{u}_\infty, \mathbf{v}_\infty) &= \nabla_{\mathbf{z}} \Phi_m(\mathbf{u}_\infty, \mathbf{v}_{p,\infty}) \\ &= \nabla_{\mathbf{z}} \Phi(\mathbf{u}_\infty, \mathbf{v}_{p,\infty}) + \boldsymbol{\lambda}_\infty^\Phi \\ &= \nabla_{\mathbf{z}_p} \Phi_p(\mathbf{u}_\infty, \mathbf{v}_{p,\infty}). \end{aligned} \quad (\text{C.10c})$$

Similarly, for interconnection equation $\mathbf{Q}_m(\mathbf{u}, \mathbf{v})$ in (C.8) i.e. (4.5d)

$$\nabla_{\mathbf{z}} \mathbf{Q}_m(\mathbf{u}_\infty, \mathbf{v}_\infty) = \nabla_{\mathbf{z}_p} \mathbf{Q}_p(\mathbf{u}_\infty, \mathbf{v}_{p,\infty}). \quad (\text{C.10d})$$

where $\mathbf{Q}_p := \mathbf{v}_p - \mathbf{H}_p(\mathbf{y}_p)$. Hence, upon convergence, we have KKT matching in Problems (4.1) and (4.5). Since, by Assumption 4.1, $(\mathbf{u}_\infty, \mathbf{v}_\infty) = (\mathbf{u}_\infty, \mathbf{v}_{p,\infty})$ is a KKT point of Problem (4.5), it is also a KKT point of (4.1) (Marchetti et al., 2009, Thm. 1). \square

C.5 Proof of Proposition 4.3

Proof. Part (i): The iterative procedure of Steps (a) and (c) in Algorithm 4.3 are essentially a dual decomposition (Necoara et al., 2011; Bertsekas and Tsitsiklis, 1997). Due to Assumption 4.4, Problem (4.13) is strictly convex. Hence, convergence of the inner loop of Algorithm 4.3 to the optimal solution is discussed in (Necoara et al., 2011, Thm. 3.5).

Part (ii): For the sake of simplicity, the following notation is used:

$$\begin{aligned}
\Phi_{p,i,k} &:= \Phi_{p,i}(\mathbf{u}_{i,k}, \mathbf{v}_{p,i,k}) \\
\Phi_{i,k} &:= \Phi_i(\mathbf{u}_{i,k}, \mathbf{v}_{i,k}) \\
\Phi_k &:= \Phi(\mathbf{u}_k, \mathbf{v}_k) \\
\mathbf{F}_{p,k} &:= \mathbf{F}_p(\mathbf{u}_k, \mathbf{v}_{p,k}) \\
\mathbf{F}_k &:= \mathbf{F}(\mathbf{u}_k, \mathbf{v}_k) \\
\mathbf{z}_{p,i,k} &:= [\mathbf{u}_{i,k}^\top, \mathbf{v}_{p,i,k}^\top]^\top \\
\mathbf{z}_{i,k} &:= [\mathbf{u}_{i,k}^\top, \mathbf{v}_{i,k}^\top]^\top \\
\phi_{p,k} &:= \phi_p(\mathbf{u}_k) \\
\mathbf{z}^\top &:= [\mathbf{z}_1^\top, \dots, \mathbf{z}_N^\top] \\
\mathbf{z}_p^\top &:= [\mathbf{z}_{p,1}^\top, \dots, \mathbf{z}_{p,N}^\top].
\end{aligned}$$

The proof is divided in 4 separate steps. Namely, the relation between plant cost function value at points \mathbf{u}_k and \mathbf{u}_{k+1} is provided based on the properties of the model and plant cost function. In the first step, the relation between model costs $\Phi(\mathbf{z}_{k+1})$, $\Phi(\mathbf{z}_{p,k})$ and plant cost gradient $\nabla_{\mathbf{z}}\Phi(\mathbf{z}_{p,k})$ is established. Then in the second step, the model cost entities are eliminated. In the third step, relation between model interconnection variables and input at points \mathbf{u}_k and \mathbf{u}_{k+1} is derived. The step 4 combines conditions from previous steps to finalize the proof.

Step 1: Here, a relationship between the model costs $\Phi(\mathbf{z}_{k+1})$ and $\Phi(\mathbf{z}_{p,k})$ is derived by expanding the functions $\Phi_{i,k}^U(\mathbf{u}_i, \mathbf{v}_i)$, $\forall i \in \mathcal{N}$ in (4.11).

Namely, upon convergence of the inner loop to the model optimum corresponding to $\mathbf{z}_{i,k+1}^\top = [\mathbf{u}_{i,k+1}^\top, \mathbf{v}_{i,k+1}^\top]$, $\forall i \in \mathcal{N}$ and dual variable $\boldsymbol{\mu}$, the following holds:

$$\begin{aligned}
\sum_{i \in \mathcal{N}} \left(\Phi_i(\mathbf{u}_{i,k+1}, \mathbf{v}_{i,k+1}) + \left(\boldsymbol{\lambda}_k^{\Phi_i} \right)^\top (\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}) + \frac{\delta^{\Phi_i}}{2} \|\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}\|_2^2 \right) + \boldsymbol{\mu}^\top (\mathbf{v}_{k+1} - \mathbf{H}\mathbf{y}_{m,k+1}) \leq \\
\sum_{i \in \mathcal{N}} \left(\Phi_i(\mathbf{u}_i, \mathbf{v}_i) + \left(\boldsymbol{\lambda}_k^{\Phi_i} \right)^\top (\mathbf{z}_i - \mathbf{z}_{p,i,k}) + \frac{\delta^{\Phi_i}}{2} \|\mathbf{z}_i - \mathbf{z}_{p,i,k}\|_2^2 \right) + \boldsymbol{\mu}^\top (\mathbf{v} - \mathbf{H}\mathbf{y}_m(\mathbf{u}, \mathbf{v})) \quad (\text{C.12a})
\end{aligned}$$

By definition, convergence of inner loop ensures feasibility of the interconnection equation for the modified model, i.e.

$$\mathbf{v}_{k+1} - \mathbf{H}\mathbf{y}_{m,k+1} = \mathbf{0}. \quad (\text{C.12b})$$

Appendix C. Proofs

Moreover, by evaluating (C.12a) at point $\mathbf{z} = \mathbf{z}_{p,k}$, one obtains

$$\sum_{i \in \mathcal{N}} \left(\Phi_i(\mathbf{u}_{i,k+1}, \mathbf{v}_{i,k+1}) + \left(\boldsymbol{\lambda}_k^{\Phi_i} \right)^\top (\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}) + \frac{\delta^{\Phi_i}}{2} \|\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}\|_2^2 \right) \leq \sum_{i \in \mathcal{N}} \Phi_i(\mathbf{u}_{p,i,k}, \mathbf{v}_{p,i,k}). \quad (\text{C.12c})$$

Hence, utilizing

$$\begin{aligned} \Phi_{k+1} &= \sum_{i \in \mathcal{N}} \Phi_i(\mathbf{u}_{i,k+1}, \mathbf{v}_{i,k+1}) \\ \Phi(\mathbf{z}_{p,k}) &= \sum_{i \in \mathcal{N}} \Phi_i(\mathbf{u}_{p,i,k}, \mathbf{v}_{p,i,k}) \\ \left(\boldsymbol{\lambda}_k^{\Phi_i} \right)^\top &= \nabla_{\mathbf{z}_{p,i}} \Phi_{p,i,k} - \nabla_{\mathbf{z}_i} \Phi_i(\mathbf{z}_{p,i,k}) \end{aligned}$$

in (C.12c) yields:

$$\Phi_{k+1} + \sum_{i \in \mathcal{N}} \left(\nabla_{\mathbf{z}_{p,i}} \Phi_{p,i,k} - \nabla_{\mathbf{z}_i} \Phi_i(\mathbf{z}_{p,i,k}) \right)^\top (\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}) + \sum_{i \in \mathcal{N}} \frac{\delta^{\Phi_i}}{2} \|\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}\|_2^2 \leq \Phi(\mathbf{z}_{p,k}). \quad (\text{C.12d})$$

Step 2: Now, the result of Step (1) i.e. (C.12d) and properties of the model cost function are used to give guarantees on the value of the plant gradient at the new point \mathbf{u}_{k+1} . We start from the fact that Assumption 4.5 implies

$$\Phi(\mathbf{z}_{p,k}) \leq \Phi_{k+1} + \sum_{i \in \mathcal{N}} \nabla_{\mathbf{z}_i} \Phi_{i,k+1}^\top (\mathbf{z}_{p,i,k} - \mathbf{z}_{i,k+1}) + \sum_{i \in \mathcal{N}} \frac{\delta_i^{\Phi_m}}{2} \|\mathbf{z}_{p,i,k} - \mathbf{z}_{i,k+1}\|_2^2. \quad (\text{C.12e})$$

Hence, summing up (C.12d) and (C.12e) gives:

$$\begin{aligned} \sum_{i \in \mathcal{N}} \nabla_{\mathbf{z}_{p,i}} \Phi_{p,i,k}^\top (\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}) + \sum_{i \in \mathcal{N}} \left(\nabla_{\mathbf{z}_i} \Phi_{i,k+1} - \nabla_{\mathbf{z}_i} \Phi_i(\mathbf{z}_{p,i,k}) \right)^\top (\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}) \\ + \sum_{i \in \mathcal{N}} \frac{\delta^{\Phi_i} - \delta_i^{\Phi_m}}{2} \|\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}\|_2^2 \leq 0. \quad (\text{C.12f}) \end{aligned}$$

In order to cancel the second term of the above inequality, the Cauchy-Schwarz inequality is used and the Lipschitz Assumption 4.5 to derive the following expression:

$$\begin{aligned} - \sum_{i \in \mathcal{N}} \left(\nabla_{\mathbf{z}_i} \Phi_{i,k+1} - \nabla_{\mathbf{z}_i} \Phi_i(\mathbf{z}_{p,i,k}) \right)^\top (\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}) \\ \leq \sum_{i \in \mathcal{N}} \|\nabla_{\mathbf{z}_i} \Phi_{i,k+1} - \nabla_{\mathbf{z}_i} \Phi_i(\mathbf{z}_{p,i,k})\|_2 \|\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}\|_2 \\ \leq \sum_{i \in \mathcal{N}} \delta_i^{\Phi_m} \|\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}\|_2^2. \quad (\text{C.12g}) \end{aligned}$$

Summing up (C.12f) and (C.12g) gives:

$$\sum_{i \in \mathcal{N}} \nabla_{\mathbf{z}_{p,i}} \Phi_{p,i,k}^\top (\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}) + \sum_{i \in \mathcal{N}} \frac{\delta^{\Phi_i} - 3\delta_i^{\Phi_m}}{2} \|\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}\|_2^2 \leq 0. \quad (\text{C.12h})$$

Finally, simplification of (C.12h) yields

$$\nabla_{\mathbf{z}_p} \Phi_{p,k}^\top (\mathbf{z}_{k+1} - \mathbf{z}_{p,k}) + \sum_{i \in \mathcal{N}} \frac{\delta^{\Phi_i} - 3\delta_i^{\Phi_m}}{2} \|\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}\|_2^2 \leq 0 \quad (\text{C.12i})$$

$$\nabla_{\mathbf{u}} \Phi_{p,k}^\top (\mathbf{u}_{k+1} - \mathbf{u}_k) + \nabla_{\mathbf{v}_p} \Phi_{p,k}^\top (\mathbf{v}_{k+1} - \mathbf{v}_{p,k}) + \sum_{i \in \mathcal{N}} \frac{\delta^{\Phi_i} - 3\delta_i^{\Phi_m}}{2} \|\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}\|_2^2 \leq 0. \quad (\text{C.12j})$$

Step 3: Now, $\mathbf{v}_{k+1} - \mathbf{v}_{p,k}$ is expressed as a function of $\mathbf{u}_{k+1} - \mathbf{u}_k$. Recall from (C.12b) that

$$\begin{aligned} \mathbf{v}_{k+1} &= \mathbf{H} \left(\mathbf{F}_{k+1} + \boldsymbol{\varepsilon}_k^F + \left(\boldsymbol{\lambda}_k^F \right)^\top (\mathbf{z}_{k+1} - \mathbf{z}_{p,k}) \right) \\ &= \mathbf{H} \left(\mathbf{F}_{k+1} + \mathbf{F}_{p,k} - \mathbf{F}(\mathbf{z}_{p,k}) + \left(\nabla_{\mathbf{z}_p} \mathbf{F}_{p,k} - \nabla_{\mathbf{z}} \mathbf{F}(\mathbf{z}_{p,k}) \right)^\top (\mathbf{z}_{k+1} - \mathbf{z}_{p,k}) \right). \end{aligned} \quad (\text{C.13a})$$

According to Assumption 4.4, the model output functions \mathbf{F} are linear in \mathbf{u} and \mathbf{v} . Hence, one can write:

$$\mathbf{F}_{k+1} = \nabla_{\mathbf{z}} \mathbf{F}^\top \mathbf{z}_{k+1} + \mathbf{F}(\mathbf{0}) \quad (\text{C.13b})$$

$$\mathbf{F}(\mathbf{z}_{p,k}) = \nabla_{\mathbf{z}} \mathbf{F}^\top \mathbf{z}_{p,k} + \mathbf{F}(\mathbf{0}). \quad (\text{C.13c})$$

Substituting (C.13b) and (C.13c) in (C.13a) yields

$$\mathbf{v}_{k+1} = \mathbf{H} \left(\mathbf{F}_{p,k} + \nabla_{\mathbf{z}_p} \mathbf{F}_{p,k}^\top (\mathbf{z}_{k+1} - \mathbf{z}_{p,k}) \right) \quad (\text{C.13d})$$

$$= \mathbf{v}_{p,k} + \mathbf{H} \nabla_{\mathbf{z}_p} \mathbf{F}_{p,k}^\top (\mathbf{z}_{k+1} - \mathbf{z}_{p,k}) \quad (\text{C.13e})$$

$$= \mathbf{v}_{p,k} + \mathbf{H} \nabla_{\mathbf{u}} \mathbf{F}_{p,k}^\top (\mathbf{u}_{k+1} - \mathbf{u}_k) + \mathbf{H} \nabla_{\mathbf{v}_p} \mathbf{F}_{p,k}^\top (\mathbf{v}_{k+1} - \mathbf{v}_{p,k}). \quad (\text{C.13f})$$

Hence, the following equality is obtained by rearranging (C.13d)

$$\left(\mathbf{I} - \mathbf{H} \nabla_{\mathbf{v}_p} \mathbf{F}_{p,k}^\top \right) (\mathbf{v}_{k+1} - \mathbf{v}_{p,k}) = \mathbf{H} \nabla_{\mathbf{u}} \mathbf{F}_{p,k}^\top (\mathbf{u}_{k+1} - \mathbf{u}_k).$$

Due to Assumption 4.6, the matrix $\left(\mathbf{I} - \mathbf{H} \nabla_{\mathbf{v}_p} \mathbf{F}_{p,k}^\top \right)$ is invertible. Hence,

$$\mathbf{v}_{k+1} - \mathbf{v}_{p,k} = \left(\mathbf{I} - \mathbf{H} \nabla_{\mathbf{v}_p} \mathbf{F}_{p,k}^\top \right)^{-1} \mathbf{H} \nabla_{\mathbf{u}} \mathbf{F}_{p,k}^\top (\mathbf{u}_{k+1} - \mathbf{u}_k). \quad (\text{C.13g})$$

According to Assumption 3.1 and (4.14a), we have $\mathbf{h}_p = \mathbf{v}_p$. Differentiating the plant interconnection variables $\mathbf{h}_p = \mathbf{v}_p = \mathbf{H} \mathbf{y}_p$ in (4.1d) at the point \mathbf{u}_k gives:

$$\frac{d\mathbf{v}_{p,k}}{d\mathbf{u}} = \nabla_{\mathbf{u}} \mathbf{h}_{p,k}^\top = \mathbf{H} \left(\nabla_{\mathbf{u}} \mathbf{F}_{p,k}^\top + \nabla_{\mathbf{v}_p} \mathbf{F}_{p,k}^\top \nabla_{\mathbf{h}} \mathbf{v}_{p,k}^\top \nabla_{\mathbf{u}} \mathbf{h}_{p,k}^\top \right). \quad (\text{C.13h})$$

Appendix C. Proofs

Since $\nabla_{\mathbf{h}} \mathbf{v}_{p,k}^\top = \mathbf{I}$, then by solving (C.13h) for $\nabla_{\mathbf{u}} \mathbf{h}_{p,k}^\top$, one obtains:

$$\nabla_{\mathbf{u}} \mathbf{h}_{p,k}^\top = \left(\mathbf{I} - \mathbf{H} \nabla_{\mathbf{v}_p} \mathbf{F}_{p,k}^\top \right)^{-1} \mathbf{H} \nabla_{\mathbf{u}} \mathbf{F}_{p,k}^\top. \quad (\text{C.13i})$$

Hence, substituting (C.13i) in (C.13g) gives:

$$\mathbf{v}_{k+1} - \mathbf{v}_{p,k} = \nabla_{\mathbf{u}} \mathbf{h}_{p,k}^\top (\mathbf{u}_{k+1} - \mathbf{u}_k). \quad (\text{C.13j})$$

Step 4: Finally, the results from the previous two steps are used to show the monotonic cost decrease condition. In particular, substituting (C.13j) in (C.12j) gives:

$$\nabla_{\mathbf{u}} \Phi_{p,k}^\top (\mathbf{u}_{k+1} - \mathbf{u}_k) + \nabla_{\mathbf{v}_p} \Phi_{p,k}^\top \nabla_{\mathbf{u}} \mathbf{h}_{p,k}^\top (\mathbf{u}_{k+1} - \mathbf{u}_k) + \sum_{i \in \mathcal{N}} \frac{\delta^{\Phi_i} - 3\delta_i^{\Phi_m}}{2} \|\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}\|_2^2 \leq 0,$$

or

$$\left(\nabla_{\mathbf{u}} \Phi_{p,k}^\top + \nabla_{\mathbf{v}_p} \Phi_{p,k}^\top \nabla_{\mathbf{u}} \mathbf{h}_{p,k}^\top \right) (\mathbf{u}_{k+1} - \mathbf{u}_k) + \sum_{i \in \mathcal{N}} \frac{\delta^{\Phi_i} - 3\delta_i^{\Phi_m}}{2} \|\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}\|_2^2 \leq 0. \quad (\text{C.14a})$$

Since $\nabla_{\mathbf{u}} \Phi_{p,k}^\top + \nabla_{\mathbf{v}_p} \Phi_{p,k}^\top \nabla_{\mathbf{u}} \mathbf{h}_{p,k}^\top = \frac{d}{d\mathbf{u}} \Phi_{p,k}$, one obtains:

$$\frac{d}{d\mathbf{u}} \Phi_{p,k} (\mathbf{u}_{k+1} - \mathbf{u}_k) + \sum_{i \in \mathcal{N}} \frac{\delta^{\Phi_i} - 3\delta_i^{\Phi_m}}{2} \|\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}\|_2^2 \leq 0. \quad (\text{C.14b})$$

Expanding inequality (C.14b) and using that $\frac{d}{d\mathbf{u}} \Phi_{p,k} = \nabla_{\mathbf{u}} \phi_{p,k}^\top$ from (4.14c)-(4.14d) leads to:

$$\nabla_{\mathbf{u}} \phi_{p,k}^\top (\mathbf{u}_{k+1} - \mathbf{u}_k) + \sum_{i \in \mathcal{N}} \frac{\delta^{\Phi_i} - 3\delta_i^{\Phi_m}}{2} \|\mathbf{u}_{i,k+1} - \mathbf{u}_{i,k}\|_2^2 + \sum_{i \in \mathcal{N}} \frac{\delta^{\Phi_i} - 3\delta_i^{\Phi_m}}{2} \|\mathbf{v}_{i,k+1} - \mathbf{v}_{p,i,k}\|_2^2 \leq 0. \quad (\text{C.14c})$$

Assumption 4.5 implies:

$$\phi_p(\mathbf{u}_{k+1}) \leq \phi_p(\mathbf{u}_k) + \nabla_{\mathbf{u}} \phi_{p,k}^\top (\mathbf{u}_{k+1} - \mathbf{u}_k) + \frac{\delta^{\phi_p}}{2} \|\mathbf{u}_{i,k+1} - \mathbf{u}_{i,k}\|_2^2. \quad (\text{C.15a})$$

Summing up (C.14c) and (C.15a) gives the following inequality:

$$\begin{aligned} \phi_p(\mathbf{u}_{k+1}) - \phi_p(\mathbf{u}_k) &\leq \\ &- \sum_{i \in \mathcal{N}} \frac{\delta^{\Phi_i} - 3\delta_i^{\Phi_m} - \delta^{\phi_p}}{2} \|\mathbf{u}_{i,k+1} - \mathbf{u}_{p,i,k}\|_2^2 - \sum_{i \in \mathcal{N}} \frac{\delta^{\Phi_i} - 3\delta_i^{\Phi_m}}{2} \|\mathbf{v}_{i,k+1} - \mathbf{v}_{p,i,k}\|_2^2. \end{aligned} \quad (\text{C.15b})$$

Hence, the plant cost decreases if

$$\delta^{\Phi_i} - 3\delta_i^{\Phi_m} - \delta^{\phi_p} \geq 0, \forall i \in \mathcal{N}.$$

This condition guarantees monotonic decrease of the plant cost. In the following part of the proof, the plant feasibility issue will be addressed.

Part (iii): We start by evaluating (4.11) at the point $\mathbf{z}_{i,k+1}$:

$$G_{i,j}(\mathbf{z}_{i,k+1}) + \boldsymbol{\varepsilon}_k^{G_{i,j}} + \left(\boldsymbol{\lambda}_k^{G_{i,j}}\right)^\top (\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}) + \frac{\delta_{i,j}^G}{2} \|\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}\|_2^2 \leq 0, \quad j = 1, \dots, n_{G_i}. \quad (\text{C.16})$$

Upon expanding the modifiers, we obtain:

$$G_{i,j}(\mathbf{z}_{i,k+1}) + G_{p,i,j}(\mathbf{z}_{p,i,k}) - G_{i,j}(\mathbf{z}_{p,i,k}) + \left(\nabla_{\mathbf{z}_{p,i}} G_{p,i,j}^\top(\mathbf{z}_{p,i,k}) - \nabla_{\mathbf{z}_i} G_{i,j}^\top(\mathbf{z}_{p,i,k})\right) (\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}) + \frac{\delta_{i,j}^G}{2} \|\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}\|_2^2 \leq 0. \quad (\text{C.17a})$$

Since, from Assumption 4.4, the model inequality constraints are linear,

$$G_{i,j}(\mathbf{z}_{i,k+1}) = \nabla_{\mathbf{z}_i} G_{i,j}^\top \mathbf{z}_{i,k+1} + G_{i,j}(0) \quad (\text{C.17b})$$

$$G_{i,j}(\mathbf{z}_{p,i,k}) = \nabla_{\mathbf{z}_i} G_{i,j}^\top \mathbf{z}_{p,i,k} + G_{i,j}(0), \quad (\text{C.17c})$$

substituting (C.17b) and (C.17c) in (C.17a) gives:

$$G_{p,i,j}(\mathbf{z}_{p,i,k}) + \nabla_{\mathbf{z}_{p,i}} G_{p,i,j}^\top(\mathbf{z}_{p,i,k}) (\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}) + \frac{\delta_{i,j}^G}{2} \|\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}\|_2^2 \leq 0. \quad (\text{C.17d})$$

Furthermore, expanding the second term of the previous inequality yields

$$\nabla_{\mathbf{z}_i} G_{p,i,j}^\top(\mathbf{z}_{p,i,k}) (\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}) = \nabla_{\mathbf{u}_i} G_{p,i,j,k}^\top(\mathbf{u}_{i,k+1} - \mathbf{u}_{i,k}) + \nabla_{\mathbf{v}_i} G_{p,i,j,k}^\top(\mathbf{v}_{i,k+1} - \mathbf{v}_{p,i,k}). \quad (\text{C.17e})$$

The i -th row of (C.13j) can be written as:

$$\mathbf{v}_{i,k+1} - \mathbf{v}_{p,i,k} = \nabla_{\mathbf{u}} \mathbf{h}_{p,i,k}^\top (\mathbf{u}_{k+1} - \mathbf{u}_k), \quad (\text{C.17f})$$

where $\nabla_{\mathbf{u}} \mathbf{h}_{p,i,k}^\top$ is the i^{th} block-row matrix of $\nabla_{\mathbf{u}} \mathbf{h}_{p,k}^\top$. Next, substituting (C.17f) in (C.17e) gives:

$$\begin{aligned} \nabla_{\mathbf{z}_i} G_{p,i,j}^\top(\mathbf{z}_{p,i,k}) (\mathbf{z}_{i,k+1} - \mathbf{z}_{p,i,k}) &= \nabla_{\mathbf{u}_i} G_{p,i,j,k}^\top \mathbf{I}_{\mathbf{u}_i} (\mathbf{u}_{k+1} - \mathbf{u}_k) + \nabla_{\mathbf{v}_i} G_{p,i,j,k}^\top \nabla_{\mathbf{u}} \mathbf{h}_{p,i,k}^\top (\mathbf{u}_{k+1} - \mathbf{u}_k) \\ &= \left(\nabla_{\mathbf{u}_i} G_{p,i,j,k}^\top \mathbf{I}_{\mathbf{u}_i} + \nabla_{\mathbf{v}_i} G_{p,i,j,k}^\top \nabla_{\mathbf{u}} \mathbf{h}_{p,i,k}^\top \right) (\mathbf{u}_{k+1} - \mathbf{u}_k) \\ &= \frac{dG_{p,i,j,k}}{d\mathbf{u}} (\mathbf{u}_{k+1} - \mathbf{u}_k) \\ &= \nabla_{\mathbf{u}} \mathbf{g}_{p,i,j,k}^\top (\mathbf{u}_{k+1} - \mathbf{u}_k). \end{aligned} \quad (\text{C.17g})$$

Here we introduced the selection matrix $\mathbf{I}_{\mathbf{u}_i} = \frac{d\mathbf{u}_i}{d\mathbf{u}} \in \mathbb{R}^{n_{u_i} \times n_u}$ and we used the fact that $\nabla_{\mathbf{h}_i} \mathbf{v}_{p,i,k}^\top = \mathbf{I}$. For the last two lines, we also used that $G_{p,i,j}(\mathbf{z}_{p,i,k}) = \mathbf{g}_{p,i,j}(\mathbf{u}_k)$ (from Assumption 3.1 and

Appendix C. Proofs

(4.14b)). Substituting (C.17g) in (C.17d), for all $i \in \mathcal{N}$, gives:

$$g_{p,i,j}(\mathbf{u}_k) + \nabla_{\mathbf{u}} g_{p,i,k}^\top(\mathbf{u}_{k+1} - \mathbf{u}_k) + \frac{\delta_{i,j}^G}{2} (\|\mathbf{u}_{i,k+1} - \mathbf{u}_{i,k}\|_2^2 + \|\mathbf{v}_{i,k+1} - \mathbf{v}_{p,i,k}\|_2^2) \leq 0. \quad (\text{C.17h})$$

Lipschitz continuity of the gradient of $g_{p,i,j}$ (Assumption 4.5) implies

$$g_{p,i,j}(\mathbf{u}_{k+1}) \leq g_{p,i,j}(\mathbf{u}_k) + \nabla_{\mathbf{u}} g_{p,i,j,k}^\top(\mathbf{u}_{k+1} - \mathbf{u}_k) + \frac{\delta_{i,j}^{g_p}}{2} \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2^2. \quad (\text{C.18})$$

Combining (C.17h) and (C.18) yields

$$g_{p,i,j}(\mathbf{u}_{k+1}) \leq \frac{\delta_{i,j}^{g_p}}{2} \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2^2 - \frac{\delta_{i,j}^G}{2} (\|\mathbf{u}_{i,k+1} - \mathbf{u}_{i,k}\|_2^2 + \|\mathbf{v}_{i,k+1} - \mathbf{v}_{p,i,k}\|_2^2). \quad (\text{C.19})$$

Finally, using (C.17f) gives

$$g_{p,i,j}(\mathbf{u}_{k+1}) \leq \frac{\delta_{i,j}^{g_p}}{2} \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2^2 - \frac{\delta_{i,j}^G}{2} \left(\|\mathbf{u}_{i,k+1} - \mathbf{u}_{i,k}\|_2^2 + \|\nabla_{\mathbf{u}} \mathbf{h}_{p,i,k}^\top\|_2^2 \|\mathbf{u}_{k+1} - \mathbf{u}_k\|_2^2 \right). \quad (\text{C.20})$$

It remains to discuss two possible cases. The first one is the case with $\|\nabla_{\mathbf{u}} \mathbf{h}_{p,i,k}^\top\|_2 = 0$, which means that the constraint $g_{p,i,j}(\mathbf{u}) = G_{p,i,j}(\mathbf{u}_i, \mathbf{v}_{p,i})$ is only a function of the local manipulated variable \mathbf{u}_i . The analysis is the same as in the centralized case (Marchetti et al., 2017), and feasibility is guaranteed if $\delta_{i,j}^G > \delta_{i,j}^{g_p}$.

In the second case is given by $\|\nabla_{\mathbf{u}} \mathbf{h}_{p,i,k}^\top\|_2 \neq 0$, the right-hand side of (C.20) is negative if $\delta_{i,j}^G > \delta_{i,j}^{g_p} / \|\nabla_{\mathbf{u}} \mathbf{h}_{p,i,k}^\top\|_2^2$. Hence, feasibility is ensured at each iterate k if $\delta_{i,j}^G > \delta_{i,j}^\Delta$, where $\delta_{i,j}^\Delta$ is defined as

$$\delta_{i,j}^\Delta = \begin{cases} \delta_{i,j}^{g_p} & \text{if } \|\nabla_{\mathbf{u}} \mathbf{h}_{p,i,k}^\top\|_2 = 0 \\ \delta_{i,j}^{g_p} / \|\nabla_{\mathbf{u}} \mathbf{h}_{p,i,k}^\top\|_2^2 & \text{if } \|\nabla_{\mathbf{u}} \mathbf{h}_{p,i,k}^\top\|_2 \neq 0. \end{cases} \quad (\text{C.21})$$

□

Bibliography

- Naturalgas.org web site. <<http://naturalgas.org/>>, 2017.
- Bundesministerium für Wirtschaft und Energie, Drucksache 18/11518, March 14 2017. URL <http://dipbt.bundestag.de/doc/btd/18/115/1811518.pdf>.
- M. Abbaspour, K. S. Chapman, and P. Krishnaswami. Nonisothermal compressor station optimization. *Journal of energy resources technology*, 127(2):131–141, 2005.
- M. Agarwal et al. Feasibility of on-line reoptimization in batch processes. *Chemical Engineering Communications*, 158:19–29, 1997.
- W. Al-Gherwi, H. Budman, and A. Elkamel. A robust distributed model predictive control algorithm. *J. Process Contr.*, 21(8):1127–1137, 2011. ISSN 0959-1524.
- V. Alstad and S. Skogestad. Null space method for selecting optimal measurement combinations as controlled variables. *Industrial & Engineering Chemistry Research*, 46(3):846–853, 2007.
- M. S. Bazaraa, H. D. Sherali, and C. M. Shetty. *Nonlinear Programming: Theory and Algorithms*. John Wiley and Sons, New Jersey, 3rd edition, 2006.
- D. P. Bertsekas and J. N. Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, Belmont, Massachusetts, 1997.
- H. P. Bloch. *A Practical Guide to Compressor Technology*. John Wiley & Sons, 2006.
- J. Bolte, Sh. Sabach, and M. Teboulle. Proximal alternating linearized minimization for non-convex and nonsmooth problems. *Mathematical Programming*, 146(1-2):459–494, 2014.
- C. Borraz-Sánchez and D. Haugland. Optimization methods for pipeline transportation of natural gas with variable specific gravity and compressibility. *Top*, 21(3):524–541, 2013.
- G. E. P. Box and N. R. Draper. *Evolutionary Operation. A Statistical Method for Process Improvement*. John Wiley, New York, 1969.
- M. P. Boyce. *Centrifugal Compressors: A Basic Guide*. PennWell Books, 2003.

Bibliography

- S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.
- M. Brdyś and P. Tatjewski. An algorithm for steady-state optimizing dual control of uncertain plants. In *Proc. 1st IFAC Workshop on New Trends in Design of Control Systems*, pages 249–254, Smolenice, Slovakia, 1994.
- M. Brdyś and P. Tatjewski. *Iterative Algorithms for Multilayer Optimizing Control*. Imperial College Press, London UK, 2005.
- M. Brdys, N. Abdullah, and P. D. Roberts. On augmented model-based double iterative loop techniques: derivation, applicability and convergence'. *Research Memorandum CEC-MB/NA/PDR-54*, Control Engineering Centre, City University, London, UK, 1987.
- M. A. Brdys, N. Abdullah, and P. D. Roberts. Hierarchical adaptive techniques for optimizing control of large-scale steady-state systems: Optimality, iterative strategies, and their convergence. *IMA J. of Mathematical Control and Information*, 7(3):199–233, 1990.
- R. Brekelmans, L. T. Driessen, H. Hamers, and D. Den Hertog. Gradient estimation schemes for noisy functions. *Journal of Optimization Theory and Applications*, 126(3):529–551, 2005.
- M. Bryds, P. D. Roberts, M. M. Badi, I. C. Kokkinos, and N. Abdullah. Double loop iterative strategies for hierarchical control of industrial processes. *Automatica*, 25(5):743–751, 1989.
- G. A. Bunin. On the equivalence between the modifier-adaptation and trust-region frameworks. *Comp. Chem. Eng.*, 71:154–157, 2014.
- G. A. Bunin, Z. Wuillemin, G. François, A. Nakajo, L. Tsikonis, and D. Bonvin. Experimental real-time optimization of a solid oxide fuel cell stack via constraint adaptation. In *23rd International Conference on Efficiency, Cost, Optimization, Simulation and Environmental Impact of Energy Systems*, Lausanne, Switzerland, 2010.
- G. A. Bunin, G. François, and D. Bonvin. From discrete measurements to bounded gradient estimates: A look at some regularizing structures. *Industrial & Engineering Chemistry Research*, 52(35):12500–12513, 2013.
- E. Camponogara, L. F. Nazari, and C. N. Meneses. A revised model for compressor design and scheduling in gas-lifted oil fields. *IIE Transactions*, 44(5):342–351, 2012.
- B. Chachuat, B. Srinivasan, and D. Bonvin. Adaptation strategies for real-time optimization. *Comp. Chem. Eng.*, 33:1557–1567, 2009.
- C. Y. Chen and B. Joseph. On-line optimization using a two-phase approach: An application study. *Ind. Eng. Chem. Res.*, 26:1924–1930, 1987.

- P. D. Christofides, R. Scattolini, D. Muñoz de la Peña, and J. Liu. Distributed model predictive control: A tutorial review and future research directions. *Comp. Chem. Eng.*, 51:21–41, 2013.
- M. Ciccioiti, D. P. Xenos, A. E. F. Bouaswaig, N. F. Thornhill, and R. Martinez-Botas. Online performance monitoring of industrial compressors using meanline modelling. In *ASME Turbo Expo 2014: Turbine Technical Conference and Exposition*, pages V02DT42A003–V02DT42A003. American Society of Mechanical Engineers, 2014.
- A. Cortinovis, H. J. Ferreau, D. Lewandowski, and M. Mercangöz. Experimental evaluation of mpc-based anti-surge and process control for electric driven centrifugal gas compressors. *J. Process Contr.*, 34:13–25, 2015.
- A. Cortinovis, M. Mercangöz, M. Zovadelli, D. Pareschi, A. De Marco, and S. Bittanti. Online performance tracking and load sharing optimization for parallel operation of gas compressors. *Comp. Chem. Eng.*, 88:145–156, 2016.
- S. Costello, G. François, and D. Bonvin. Real-time optimization when the plant and the model have different inputs. In *IFAC Symp. Dycops*, Mumbai, India, 2013.
- S. Costello, G. François, and D. Bonvin. A directional modifier-adaptation algorithm for real-time optimization. *J. Process Contr.*, 39:64–76, 2016.
- C. R. Cutler and R. T. Perry. Real time optimization with multivariable control is required to maximize profits. *Comp. Chem. Eng.*, 7(5):663–667, 1983.
- M. L. Darby, M. Nikolaou, J. Jones, and D. Nicholson. RTO: An overview and assessment of current practice. *J. Process Contr.*, 21:874–884, 2011.
- T. de Avila Ferreira, G. François, A. Marchetti, and D. Bonvin. Use of transient measurements for static real-time optimization. *IFAC-PapersOnLine*, 50(1):5737–5742, 2017.
- F. C. G. de Marco, G. P. Elias, et al. Fuel consumption model on natural gas compression stations driven by two-shaft gas turbine. In *PSIG Annual Meeting*. Pipeline Simulation Interest Group, 2011.
- S. L. Dixon and C. Hall. *Fluid mechanics and thermodynamics of turbomachinery*. Butterworth-Heinemann, 2013.
- T. F. Edgar, D. M. Himmelblau, T. C. Bickel, et al. Optimal design of gas transmission networks. *Society of Petroleum Engineers Journal*, 18(02):96–104, 1978.
- J. E. Ellis, C. Kambhampati, G. Sheng, and P. D. Roberts. Approaches to the optimizing control problem. *Int. J. Systems Sci.*, 19(10):1969–1985, 1988.
- M. Farina, G. Ferrari-Trecate, and R. Scattolini. Moving-horizon partition-based state estimation of large-scale systems. *Automatica*, 46(5):910–918, 2010. ISSN 0005-1098.

Bibliography

- T. Faulwasser and D. Bonvin. On the use of second-order modifiers for real-time optimization. *IFAC Proceedings Volumes*, 47(3):7622–7628, 2014.
- W. Findeisen, F. N. Bailey, M. Brdys, K. Malinowski, P. Tatjewski, and A. Wozniak. *Control and Coordination in Hierarchical Systems*. John Wiley & Sons, 1980.
- J. F. Forbes and T. E. Marlin. Design cost: A systematic approach to technology selection for model-based real-time optimization systems. *Comp. Chem. Eng.*, 20:717–734, 1996.
- J. F. Forbes, T. E. Marlin, and J. F. MacGregor. Model adequacy requirements for optimizing plant operations. *Comp. Chem. Eng.*, 18(6):497–510, 1994.
- G. François and D. Bonvin. Use of convex model approximations for real-time optimization via modifier adaptation. *Ind. Eng. Chem. Res.*, 52:11614–11625, 2013a.
- G. François and D. Bonvin. Measurement-based real-time optimization of chemical processes. In *Advances in Chemical Engineering*, volume 43, pages 1–50. Elsevier, 2013b.
- G. François, B. Srinivasan, and D. Bonvin. Use of measurements for enforcing the necessary conditions of optimality in the presence of constraints and uncertainty. *J. Process Contr.*, 15(6):701–712, 2005.
- G. François and D. Bonvin. Use of transient measurements for the optimization of steady-state performance via modifier adaptation. *Ind. Eng. Chem. Res.*, 53(13):5148–5159, 2013.
- W. Gao and S. Engell. Iterative set-point optimization of batch chromatography. *Comp. Chem. Eng.*, 29:1401–1409, 2005.
- W. Gao, S. Wenzel, and S. Engell. A reliable modifier-adaptation strategy for real-time optimization. *Comp. Chem. Eng.*, 91:318–328, 2016.
- A. Gopalakrishnan and L. T. Biegler. Economic nonlinear model predictive control for periodic optimal operation of gas pipeline networks. *Comp. Chem. Eng.*, 52:90–99, 2013.
- J. Gu and B. Wan. Steady state hierarchical optimizing control for large-scale industrial processes with fuzzy parameters. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 31(3):352–360, 2001.
- I. Han, C. Han, and C. Chung. Optimization of the air-and gas-supply network of a chemical plant. *Chemical Engineering Research and Design*, 82(10):1337–1343, 2004.
- M. M. F. Hasan, M. Sh. Razib, and I. A. Karimi. Optimization of compressor networks in lng operations. In *Computer Aided Chemical Engineering*, volume 27, pages 1767–1772. Elsevier, 2009.
- A. Hawryluk, K. K. Botros, H. Golshan, and B. Huynh. Multi-objective optimization of natural gas compression power train with genetic algorithms. In *2010 8th International Pipeline Conference*, pages 421–435. American Society of Mechanical Engineers, 2010.

-
- A. C. Hax. *Natural gas transmission system optimization: a mathematical programming model*. University of California, 1967.
- J.-H. Hours and C. N. Jones. A parametric nonconvex decomposition algorithm for real-time and distributed NMPC. *IEEE Transactions on Automatic Control*, 61(2):287–302, 2016.
- B. Houska, J. Frasch, and M. Diehl. An augmented Lagrangian based algorithm for distributed nonconvex optimization. *SIAM Journal on Optimization*, 26(2):1101–1127, 2016.
- Mi. Krstić and H.-H. Wang. Stability of extremum seeking feedback for general nonlinear dynamic systems. *Automatica*, 36(4):595–601, 2000.
- S. Kumar and A. Cortinovis. Load sharing optimization for parallel and serial compressor stations. In *Control Technology and Applications (CTA), 2017 IEEE Conference on*, pages 499–504. IEEE, 2017.
- R. Kurz and K. Brun. Assessment of compressors in gas storage applications. *Journal of Engineering for Gas Turbines and Power*, 132(6):062402, 2010.
- R. Kurz and K. Brun. Fouling mechanisms in axial compressors. *Journal of Engineering for Gas Turbines and Power*, 134(3):032401, 2012a.
- R. Kurz and K. Brun. Upstream and midstream compression applications: Part 1—applications. In *ASME Turbo Expo 2012: Turbine Technical Conference and Exposition*, pages 11–21. American Society of Mechanical Engineers, 2012b.
- R. Kurz, M. Lubomirsky, and K. Brun. Gas compressor station economic optimization. *International Journal of Rotating Machinery*, 2012, 2012.
- T. Larsson and S. Skogestad. Plantwide control - A review and a new design procedure. *Modeling, Identification and Control*, 21(4):209–240, 2000.
- J. Lin, Z. M. Hendawy, and P. D. Roberts. New model-based double-loop iterative strategy for integrated system optimization and parameter estimation of large-scale industrial processes. *International Journal of Control*, 47(3):753–773, 1988.
- J. Lin, L. Z. Li, B. W. Wan, and P. D. Roberts. Augmentation of several newly developed algorithms for integrated system optimization and parameter estimation of large-scale industrial processes: Convergence analysis and comparative study. *IMA J. of Mathematical Control and Information*, 6(3):333–345, 1989.
- B. Lipták. *Process Control: Instrument Engineers' Handbook*. Butterworth-Heinemann, 2013.
- J. Liu and S. J. Wright. Asynchronous stochastic coordinate descent: Parallelism and convergence properties. *SIAM Journal on Optimization*, 25(1):351–376, 2015.
- C. A. Luongo, B. J. Gilmour, and D. W. Schroeder. Optimization in natural gas transmission networks: A tool to improve operational efficiency. In *3rd SIAM Conference on Optimization*, 1989.

Bibliography

- D. Mahlke, A. Martin, and S. Moritz. A mixed integer approach for time-dependent gas network optimization. *Optimization Methods & Software*, 25(4):625–644, 2010.
- J. Mahmoudimehr and S. Sanaye. Minimization of fuel consumption of natural gas compressor stations with similar and dissimilar turbo-compressor units. *Journal of Energy Engineering*, 140(1):04013001, 2013.
- M. Mansour and J. E. Ellis. Comparison of methods for estimating real process derivatives in on-line optimization. *Applied Mathematical Modelling*, 27(4):275–291, 2003.
- A. Marchetti. A new dual modifier-adaptation approach for iterative process optimization with inaccurate models. *Comp. Chem. Eng.*, 59:89–100, 2013.
- A. Marchetti, B. Chachuat, and D. Bonvin. Modifier-adaptation methodology for real-time optimization. *Ind. Eng. Chem. Res.*, 48(13):6022–6033, 2009.
- A. Marchetti, B. Chachuat, and D. Bonvin. A dual modifier-adaptation approach for real-time optimization. *J. Process Contr.*, 20:1027–1037, 2010.
- A. Marchetti, G. François, T. Faulwasser, and D. Bonvin. Modifier adaptation for real-time optimization – Methods and applications. *Processes*, 4:1–35, 2016.
- A. Marchetti, T. Faulwasser, and D. Bonvin. A feasible-side globally convergent modifier-adaptation scheme. *J. Process Contr.*, 54:38–46, 2017.
- B. R. Marks and G. P. Wright. A general inner approximation algorithm for nonconvex mathematical programs. *Operations research*, 26(4):681–683, 1978.
- T. E. Marlin and A. N. Hrymak. Real-time operations optimization of continuous processes. In *AIChE Symposium Series - CPC-V*, volume 93, pages 156–164, 1997.
- D. Marqués and M. Morari. On-line optimization of gas pipeline networks. *Automatica*, 24(4):455–469, 1988.
- J. Martín-Aragón and M. Valdés. A method to determine the economic cost of fouling of gas turbine compressors. *Applied Thermal Engineering*, 69(1-2):261–266, 2014.
- P. Milosavljevic, A. Cortinovis, A. Marchetti, T. Faulwasser, M. Mercangöz, and D. Bonvin. Optimal load sharing of parallel compressors via modifier adaptation. In *IEEE Conference on Control Applications*, pages 1488–1493, 2016.
- P. Milosavljevic, R. Schneider, T. Faulwasser, and D. Bonvin. Distributed modifier adaptation using a coordinator and measured interconnection variables. In *20th IFAC World Congress, Toulouse, France*, 2017.
- P. Milosavljevic, A. Cortinovis, A. Marchetti, T. Faulwasser, M. Mercangöz, and D. Bonvin. Load-sharing optimization of parallel and serial compressors via modifier adaptation. *Applied Energy*, (in preparation), 2018a.

- P. Milosavljevic, A. Cortinovis, R. Schneider, T. Faulwasser, M. Mercangöz, and D. Bonvin. Optimal load sharing for serial compressors via modifier adaptation. In *European Control Conference (ECC)*, pages 2306–2311, 2018b.
- P. Milosavljevic, R. Schneider, A. Cortinovis, T. Faulwasser, and D. Bonvin. A distributed feasible-side convergent modifier-adaptation scheme for interconnected systems, with application to gas-compressor stations. *Comp. Chem. Eng.*, 115:474–486, 2018c.
- M. Mohamadi-Baghmolaei, M. Mahmoudy, D. Jafari, R. MohamadiBaghmolaei, and F. Tabkhi. Assessing and optimization of pipeline system performance using intelligent systems. *Journal of Natural Gas Science and Engineering*, 18:64–76, 2014.
- M. Morari, Y. Arkun, and G. Stephanopoulos. Studies in the synthesis of control structures for chemical processes. Part I: Formulation of the problem. Process decomposition and the classification of the control tasks. Analysis of the optimizing control structures. *AIChE J.*, 26(2):220–232, 1980.
- S. Moritz. *A Mixed Integer Approach for the Transient Case of Gas Network Optimization*. PhD thesis, Technische Universität, 2007.
- H. G. Murphy et al. Compressor performance modeling to improve efficiency and the quality of optimization decisions. In *PSIG Annual Meeting*. Pipeline Simulation Interest Group, 1989.
- D. Navia, R. Martí, D. Sarabia, G. Gutierrez, and C. de Prada. Handling infeasibilities in dual modifier-adaptation methodology for real-time optimization. *IFAC Proceedings Volumes*, 45(15):537–542, 2012.
- D. Navia, G. Gutiérrez, and C. de Prada. Nested modifier-adaptation for rto in the otto williams reactor. *IFAC Proceedings Volumes*, 46(32):123–128, 2013.
- I. Necoara, C. Savorgnan, D. Q. Tran, J. Suykens, and M. Diehl. Distributed nonlinear optimal control using sequential convex programming and smoothing techniques. In *48th IEEE Conference on Decision and Control held jointly with the 28th Chinese Control Conference*, pages 543–548, 2009.
- I. Necoara, V. Nedelcu, and I. Dumitrache. Parallel and distributed optimization methods for estimation and control in networks. *J. Process Contr.*, 21(5):756–766, 2011.
- H. H. Nguyen, V. Uraikul, C. W. Chan, and P. Tontiwachwuthikul. A comparison of automation techniques for optimization of compressor scheduling. *Advances in Engineering Software*, 39(3):178–188, 2008.
- R. Olfati-Saber, J. A. Fax, and R. M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007. ISSN 0018-9219.
- A. Osiadacz. Nonlinear programming applied to the optimum control of a gas compressor station. *International Journal for Numerical Methods in Engineering*, 15(9):1287–1301, 1980.

Bibliography

- F. Paparella, L. Domínguez, A. Cortinovis, M. Mercangöz, D. Pareschi, and S. Bittanti. Load sharing optimization of parallel compressors. In *2013 European Control Conference (ECC)*, pages 4059–4064, Zurich, Switzerland, 2013.
- A. Papasavvas, T. de Avila Ferreira, A. Marchetti, and D. Bonvin. Real-time optimization via modifier adaptation – on updating the model outputs. In *AIChE Annual Meeting*, Minneapolis, MN, 2017.
- N. Parikh and S. Boyd. Proximal algorithms. *Foundations and Trends in Optimization*, 1(3): 127–239, 2014.
- Z. Peng, Y. Xu, M. Yan, and W. Yin. Arock: an algorithmic framework for asynchronous parallel coordinate updates. *SIAM Journal on Scientific Computing*, 38(5):A2851–A2879, 2016.
- Y. Pu, M. N. Zeilinger, and C. N. Jones. Inexact fast alternating minimization algorithm for distributed model predictive control. In *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, pages 5915–5921. IEEE, 2014.
- P. N. Rao and V. N. Naikan. An optimal maintenance policy for compressor of a gas turbine power plant. *Journal of Engineering for Gas Turbines and Power*, 130(2):021801, 2008.
- P. C. Rasmussen, R. Kurz, et al. Centrifugal compressor applications-upstream and mid-stream. In *Proceedings of the 38th Turbomachinery Symposium*. Texas A&M University. Turbomachinery Laboratories, 2009.
- R. Z. Ríos-Mercado and C. Borraz-Sánchez. Optimization problems in natural gas transportation systems: A state-of-the-art review. *Applied Energy*, 147:536–555, 2015.
- P. D. Roberts. An algorithm for steady-state system optimization and parameter estimation. *J. System Science*, 10:719–734, 1979.
- P. D. Roberts. Coping with model-reality differences in industrial process optimisation - a review of integrated system optimisation and parameter estimation (ISOPE). *Computers in Industry*, 26:281–290, 1995.
- P. D. Roberts and T. W. Williams. On an algorithm for combined system optimisation and parameter estimation. *Automatica*, 17(1):199–209, 1981.
- E. A. Rodger and B. Chachuat. Design methodology of modifier adaptation for on-line optimization of uncertain processes. In *18th IFAC World Congress*, pages 4113—4118, Milano, Italy, 2010.
- R. Saidur, N. A. Rahim, and M. Hasanuzzaman. A review on compressed-air energy use and energy savings. *Renewable and Sustainable Energy Reviews*, 14(4):1135–1153, 2010.
- D. Sánchez, R. Chacartegui, J.A. Becerra, and T. Sánchez. Determining compressor wash programmes for fouled gas turbines, 2009.

- N. Sandell, P. Varaiya, M. Athans, and M. Safonov. Survey of decentralized control methods for large scale systems. *IEEE Trans. on Automatic Control*, 23(2):108–128, 1978. ISSN 0018-9286.
- R. Scattolini. Architectures for distributed and hierarchical model predictive control—A review. *J. Process Contr.*, 19(5):723–731, 2009.
- M. Schmidt, M. C. Steinbach, and B. M. Willert. High detail stationary optimization models for gas networks. *Optimization and Engineering*, 16(1):131–164, 2015.
- R. Schneider, H. Scheu, and W. Marquardt. Distributed MPC and partition-based MHE for distributed output feedback. In *19th IFAC World Congress*, pages 2183–2188, Cape Town, South Africa, 2014.
- R. Schneider, R. Hannemann-Tamás, and W. Marquardt. An iterative partition-based moving horizon estimator with coupled inequality constraints. *Automatica*, 61:302–307, 2015.
- R. Schneider, P. Milosavljevic, and D. Bonvin. Distributed modifier-adaptation schemes for the real-time optimisation of uncertain interconnected systems. *International Journal of Control*, pages 1–14, 2017.
- F. J. Serralunga, M. C. Mussati, and P. A. Aguirre. Model adaptation for real-time optimization in energy systems. *Industrial & Engineering Chemistry Research*, 52(47):16795–16810, 2013.
- F. J. Serralunga, P. A. Aguirre, and M. C. Mussati. Including disjunctions in real-time optimization. *Industrial & Engineering Chemistry Research*, 53(44):17200–17213, 2014.
- T. L. Silva and E. Camponogara. A computational analysis of multidimensional piecewise-linear models with applications to oil production optimization. *European Journal of Operational Research*, 232(3):630–642, 2014.
- M. Singhal, A. Marchetti, T. Faulwasser, and D. Bonvin. Improved directional derivatives for modifier-adaptation schemes. *IFAC-PapersOnLine*, 50(1):5718–5723, 2017.
- S. Skogestad. Self-optimizing control: The missing link between steady-state optimization and control. *Comp. Chem. Eng.*, 24:569–575, 2000.
- B. Srinivasan and D. Bonvin. Real-time optimization of batch processes via tracking the necessary conditions of optimality. *Ind. Eng. Chem. Res.*, 46(2):492–504, 2007.
- B. Srinivasan, D. Bonvin, E. Visser, and S. Palanki. Dynamic optimization of batch processes: II. Role of measurements in handling uncertainty. *Comp. Chem. Eng.*, 27:27–44, 2003.
- B. Srinivasan, L.T. Biegler, and D. Bonvin. Tracking the necessary conditions of optimality with changing set of active constraints using a barrier-penalty function. *Comp. Chem. Eng.*, 32(3):572 – 579, 2008.

Bibliography

- B. Srinivasan, G. François, and D. Bonvin. Comparison of gradient estimation methods for real-time optimization. In *Computer Aided Chemical Engineering*, volume 29, pages 607–611. Elsevier, 2011.
- N. Staroselsky and S. Mirsky. Method for controlling a multicompressor station, February 3 1987. US Patent 4,640,665.
- G. Stathopoulos and C. N. Jones. An inertial parallel and asynchronous fixed-point iteration for convex optimization. *arXiv preprint arXiv:1706.00088*, 2017.
- G. Stephanopoulos and C. Ng. Perspectives on the synthesis of plant-wide control structures. *J. Process Contr.*, 10:97–111, 2000.
- H. Sun and H. Ding. Plant simulation and operation optimisation of smr plant with different adjustment methods under part-load conditions. *Comp. Chem. Eng.*, 68:107–113, 2014.
- P. Tatjewski. Iterative optimizing set-point control - The basic principle redesigned. In *15th IFAC World Congress*, Barcelona, Spain, 2002.
- R. Tirnovan, S. Giurgea, A. Miraoui, and M. Cirrincione. Surrogate modelling of compressor characteristics for fuel-cell applications. *Applied Energy*, 85(5):394–403, 2008.
- Q. Tran-Dinh, I. Necoara, and M. Diehl. Fast inexact decomposition algorithms for large-scale separable convex optimization. *Optimization*, 65(2):325–356, 2016.
- S. A. van den Heever and I. E. Grossmann. A strategy for the integration of production planning and reactive scheduling in the optimization of a hydrogen supply network. *Comp. Chem. Eng.*, 27(12):1813–1839, 2003.
- A. N. Venkat, I. A. Hiskens, J. B. Rawlings, and S. J. Wright. Distributed MPC strategies with application to power system automatic generation control. *IEEE Trans. on Control Systems Technology*, 16(6):1192–1206, 2008. ISSN 1063-6536.
- S. Wenzel, R. Paulen, S. Krämer, B. Beisheim, and S. Engell. Price adjustment in price-based coordination using quadratic approximation. In Zdravko Kravanja, editor, *Proceedings of the 26th European Symposium on Computer Aided Process Engineering*, Portorož, Slovenia, 2016. Elsevier.
- K. N. Widell and T. Eikevik. Reducing power consumption in multi-compressor refrigeration systems. *International Journal of Refrigeration*, 33(1):88–94, 2010.
- S. Wright, C. Ditzel, M. Somani, et al. Compressor station optimization. In *PSIG Annual Meeting*. Pipeline Simulation Interest Group, 1998.
- D. P. Xenos, G. M. Kopanos, M. Ciccioiti, E. N. Pistikopoulos, and N. F. Thornhill. Operational optimization of compressors in parallel considering condition-based maintenance. In *Computer Aided Chemical Engineering*, volume 33, pages 1213–1218. Elsevier, 2014.

- D. P. Xenos, M. Ciccotti, G. M. Kopanos, A. E. F. Bouaswaig, O. Kahrs, R. Martinez-Botas, and N. F. Thornhill. Optimization of a network of compressors in parallel: Real time optimization RTO of compressors in chemical plants - An industrial case study. *Applied Energy*, 144: 51–63, 2015.
- D. P. Xenos, G. M. Kopanos, M. Ciccotti, and N. F. Thornhill. Operational optimization of networks of compressors considering condition-based maintenance. *Comp. Chem. Eng.*, 84:117–131, 2016.
- C. Xu, G. Sand, I. Harjunkoski, and S. Engell. A new heuristic for plant-wide schedule coordination problems: The intersection coordination heuristic. *Comp. Chem. Eng.*, 42:152–167, 2012.
- V. M. Zavala. Stochastic optimal control model for natural gas networks. *Comp. Chem. Eng.*, 64:103–113, 2014.
- Y. Zhang and J. F. Forbes. Performance analysis of perturbation-based methods for real-time optimization. *The Canadian J. Chem. Eng.*, 84:209–218, 2006.

Ch. des Clochetons 4bis
1004 Lausanne VD
Switzerland
☎ +41 786 934026
✉ predrag.milosavljevic@epfl.ch
Date of birth: 02.10.1987
Nationality: Serbian

Predrag Milosavljevic

Education

- 2013–present **PhD**, *École Polytechnique Fédérale de Lausanne (EPFL)*, Lausanne.
Real-Time Optimization of Interconnected Systems via Modifier Adaptation, with Application to Gas-Compressor Stations
- 2010–2011 **M.Sc in Electrical Engineering**, *Signals & Systems Department*, Belgrade.
Master thesis in the field of Process Control & Robotics: "Control synthesis of an anthropomimetic robot with passive compliance and the analysis of its energy efficiency "
- 2006–2010 **B.E.E. in Electrical Engineering**, *Signals & Systems Department*, Belgrade.
Diploma thesis in the field of Process Control & Robotics: "Control of an anthropomimetic robot arm with seven degrees of freedom"

Experience

- 2013–present **École Polytechnique Fédérale de Lausanne (EPFL)**, *Lausanne*.
 - Teaching assistant in five classes related to control theory.
 - Supervision of student projects.
- 2012 **Fraunhofer Insitute (IPK)**, *Automation Technology and Robotics Department*, Berlin.
Internship in the field of control of redundant advanced robotic systems
- 2010–2012 **Faculty of Electrical Engineering**, *Belgrade*.
 - Research assistant at European project - ECCEROBOT (Embodied Cognition in a Compliantly Engineered Robot) (www.eccerobot.org).
Development of data driven and robust control strategies.
Modeling of compliant robotic structures.

Awards and Activities

- 2012 Serbian Center for the Promotion of Science, „Days of Future: Robotics“ (24th of Sept. – 15th of Oct.) (<http://danibuducnosti.rs/>), Participant and Lecturer.
- 2012 "Ost-Ausschuss der Deutschen Wirtschaft und Deutsche Industrie - und Handelskammer" and "Zoran Djindjic Internship Programme of German Business" Scholarship for professional education in Germany.
- 2011 Award for the 1st prize at the 17th International Students Competition in Engineering – ICAMES 2011, Istanbul, Turkey.
- 2011 DAAD and IAESTE Scholarship for internship in Germany. Intern at the University of Merseburg, Germany.

Publications

- P. Milosavljevic, A. Cortinovis, A. Marchetti, T. Faulwasser, M. Mercangöz, and D. Bonvin. Load-sharing optimization of parallel and serial compressors via modifier adaptation. *Applied Energy*, (in preparation), 2018.
- P. Milosavljevic, R. Schneider, A. Cortinovis, T. Faulwasser, and D. Bonvin. A distributed feasible-side convergent modifier-adaptation scheme for interconnected systems, with application to gas-compressor stations. *Computers & Chemical Engineering*, (accepted), 2018.
- R. Schneider, P. Milosavljevic, and D. Bonvin. Accelerated and adaptive modifier-adaptation schemes for the real-time optimization of uncertain systems. *Journal of Process Control*, (submitted), 2018.
- R. Schneider, P. Milosavljevic, and D. Bonvin. Distributed modifier-adaptation schemes for the real-time optimisation of uncertain interconnected systems. *International Journal of Control*, pages 1–14, 2017.
- P. Milosavljevic, A. Cortinovis, R. Schneider, T. Faulwasser, M. Mercangöz, and D. Bonvin. Optimal load sharing for serial compressors via modifier adaptation. In *European Control Conference (ECC)*, Limassol, Cyprus (accepted), 2018.
- P. Milosavljevic, A. Cortinovis, A. Marchetti, T. Faulwasser, M. Mercangöz, and D. Bonvin. Optimal load sharing of parallel compressors via modifier adaptation. In *IEEE Conference on Control Applications*, pages 1488–1493, 2016.
- P. Milosavljevic, R. Schneider, T. Faulwasser, and D. Bonvin. Distributed modifier adaptation using a coordinator and measured interconnection variables. In *20th IFAC World Congress, Toulouse, France*, 2017.
- R. Schneider, P. Milosavljevic, and D. Bonvin. Accelerated and adaptive modifier-adaptation schemes for unconstrained uncertain systems. In *20th IFAC World Congress, Toulouse, France*, 2017.
- P. Milosavljevic, T. Faulwasser, A. Marchetti, and D. Bonvin. Time-optimal path-following operation in the presence of uncertainty. In *Control Conference (ECC), 2016 European*, pages 2228–2233. IEEE, 2016.
- S. Wittmeier, C. Alessandro, N. Bascarevic, K. Dalamagkidis, D. Devereux, A. Diamond, M. Jäntschi, K. Jovanovic, R. Knight, H. Marques, P. Milosavljevic, B. Mitra, B. Svetozarevic, V. Potkonjak, P. Pfeifer, A. Knoll, and H. Holland. Toward anthropomorphic robotics: development, simulation, and control of a musculoskeletal torso. *Artificial life*, 19(1):171–193, 2013.
- P. Milosavljevic, K. Jovanovic, N. Bascarevic, V. Potkonjak, and O. Holland. Heuristic machine-learning approach to the control of an anthropomorphic robot arm. *IFAC Proceedings Volumes*, 45(22):301–306, 2012.
- V. Potkonjak, K. Jovanovic, P. Milosavljevic, N. Bascarevic, and O. Holland. The puller-follower control concept in the multi-jointed robot body with antagonistically coupled compliant drives. In *IASTED international conference on robotics*, pages 375–381, 2011.
- N. Bascarevic, K. Jovanovic, P. Milosavljevic, V. Potkonjak, and O. Holland. Tip-over stability examination of a compliant anthropomorphic mobile robot. In *Control Applications (CCA), 2012 IEEE International Conference on*, pages 1584–1589. IEEE, 2012.

