Multiscale Concepts
in
State Estimation

– schriftliche Habilitationsleistung –

Universität Regensburg
Naturwissenschaftliche Fakultät I – Mathematik

vorgelegt von

LUISE BLANK

Regensburg, den 17. November 2003

Contents

Introduction

State estimation

Safe and economical process operations are of crucial importance in industry as well as for research in science and medicine. One can distinguish two subproblems for optimizing the process. On one hand one wishes to maximize the profit, i.e. to maximize the outcome with minimal cost while ensuring safety. This leads to the well known control problems which requires the knowledge of the current state of the process. On the other hand, there is a significant incentive to increase the understanding of the process state since any upgrade in information will enable improvement of process operations, monitoring and control. The gained information can be directly used to better manually control the process, it can be fed to some operator decision support system or to some automatic control system to lead to safer process operations of increased performance. Hence, the second subproblem is to estimate the current process states and parameters from given, typically noisy, measurements. These two problem classes have to be solved online repetitively with proceeding time (see for e.g.[6, 19, 138, 143]). The result of the state estimation serves as input for the control problem and the optimized and then implemented controls act as inputs for the estimation problem. State estimation is a backwards in time problem while control problems aim at the future. Depending on the employed problem formulations one can consider the two problems as dual to each other. In this work we concentrate on state estimation. Though we keep in mind the interaction with the corresponding control problem. Moreover, many parts of the presented work can be carried over in a straight forward way to the control problem. However, the problem of state estimation occurs also in a much wider context than as part of feedback control. Drawing conclusions from data and models is a fundamental issue and state estimation may be part of process monitoring, data reconciliation and diagnosis, inferential sensor development, and product property estimation [6].

For state estimation the underlying dynamical process is commonly described with state equations given as differential or differential algebraic equations. The known quantities are the controls and the measurements. The measurements are typically noisy and correspond in most cases to only very few states or to a certain linear combination of the states. This connection leads to the so called output equations. The functions reflecting the measurements in the process model are called output functions, or in some literature observed variables. The other states have to be accessible by the coupling in the model equations which summarize the state and the output equations. This dynamical system is the starting point for state estimation.

The first approaches for state estimation were based on linear model equations considering e.g. stationary processes. Luenberger [126, 127] introduced the terminology of an observer for a deterministic system in the sixties. This observer provides an asymptotic state estimator. When time approaches infinity the estimated state approaches the exact state [53, 121]. Independently of this approach Kalman and

Bucy [111, 112] developed an optimal stochastic observer roughly at the same time. They presume a probabilistic setting i.e. possible perturbations in the state equations and also noise in the observations which are assumed to be uncorrelated. This results into the well known and widely used Kalman filter. The optimal observer is given by employing a matrix differential Riccati equation. If the noise is a Gaussian white noise it is shown that the Kalman filter produces an estimate that has minimal mean square reconstruction error (see for e.g. [85, 118, 121]). In spite of being one of the most significant achievements in filtering theory, the Kalman filter is limited in the application of state estimation. First, successful application hinges on stochastical assumptions which are not necessarily applicable or the necessary information cannot be attained by experiments. A further drawback is its extension to nonlinear systems. The assumption of Gaussian distribution of the state variables is no longer valid when propagated through nonlinear dynamics. This difficulty is conventionally bypassed by using a local linearization and leads to the extended Kalman filter. However its performance depends on the quality of the linearized model. The extended Kalman filter can exhibit poor results and biased estimates [106, 143, 157]. Comments on nonlinear observers, for e.g. the extended Luenberger observer, can be found in [31, 143, 184]. Their application is limited to systems with certain properties (for e.g. nonlinearity with analytic functions). Another aspect is that the model equations may not be precise. Hence to obtain accurate and physically meaningful state estimates one wishes to include additional process insights for e.g. inequality constraints on some states reflecting possible maximal or minimal values of the states. This leads to the third approach of state estimation.

Satisfying inequality constraints is the domain of mathematical programming. Hence state estimation is formulated as a least squares optimization problem over the past time horizon. As time progresses the length of the time horizon increases, such that the numerical effort will finally turn out to be unacceptable. Consequently one wishes to choose a finite time interval. There exists several approaches for including the information of the previous horizon into the next one to obtain valid and stable state informations (see for e.g. [138, 144, 156, 157]). They are typically based on stochastical assumptions. Nevertheless up to now the choice of the length of the horizon is a fairly open research field. However this is not the subject of concern in this work. Given the horizon length, state estimation is formulated on a receding horizon and the optimization problem has to be solved repetitively for the moved time intervals. The formulation of state estimation as moving horizon estimation (MHE) is not new but has been proposed by numerous researchers (for a short review see [6, 143, 156]). The corresponding control problem formulation with a moving predictive time horizon (model predictive control MPC) can be seen as an historical origin for this formulation. Excellent introductions and surveys on MPC and MHE can be found in [6, 139]. In [19] an introduction into *dynamic optimization applied to moving horizons* and a summary of available numerical techniques is given.

For linear problems *moving horizon estimation* is equivalent to the Kalman filter under certain choices of the weighting matrices in the least square formulation [2, 107]. Hence a probabilistic interpretation of MHE can be the generalization of the Kalman filter. (The connection and differences between Kalman filter and MHE have been studied in recent years based on discrete time systems [144, 156, 157].) However, in real situations there is no justification for simple assumptions on the probabilistic properties of the model and measurement errors. Hence, in this work the general problem formulation we are interested in, aims at avoiding any assumptions on statistical properties of the process quantities and error functions. A *deterministic setting* is envisaged. Then the weights included into the least squares formulation are motivated by the view of state estimation as an inverse problem. They are based on the theory of ill–posed problems and their regularization. However, this does not exclude that possible stochastical knowledge can be taken into account for the choice. Another difference to most problem formulations is that we do not assume to have any information on the initial state. As a consequence we avoid possibly occurring bias due to wrongly chosen reference initial values.

In addition to the MHE approach preprocessing of the input data, i.e. of the measurements, is necessary. The discrete measurement samples are typically not given at the same instants or the same sampling rate. Also outliers can appear, signal parts can be missing and noise always arises. Using techniques of *signal analysis* the data have to be transformed into continuous representations which shall be denoised, at least to a large extent.

Due to the necessity to observe the current state of the process for safety reasons, an approximate state estimate has to be provided in a computation time limited by the process dynamic. Moreover the *online requirement* is aggravated by the a priori unknown available computational time, since several users may employ one computational resource. In addition the nonlinearity in the problem formulation rules out the estimation of the necessary computation time for obtaining a certain accuracy. Consequently the real–time view 'given an estimation accuracy compute the approximates as fast as possible' does not apply in this context. In contrast we propose to construct an algorithm which provides estimates at any instant with an approximation quality that increases correspondingly with the employed computation time. A rough estimate shall be given after a very short time and shall be improved successively as long as computation time is available. In the engineering community and in the research area of artificial intelligence such algorithms have been recently referred to as anytime algorithms [142, 160].

Multiscale approach

Choosing the MHE approach for state estimation we have to repeatedly solve online a potentially large–scale dynamic optimization problem on a receding time horizon combined with the issues of signal processing. The large scale nature as well as the real-time requirement of the problem is a clear challenge where the cutting edge of currently commercially available technology needs to be pushed further forward.

A survey of the recent state of online and real–time optimization is given in [91, 103]. For *numerical methods for dynamical optimization on a fixed time horizon* two main strategies are distinguished. The indirect methods are based on the Euler–Lagrange differential equations given by the maximum principle known in the context of calculus of variations. Then these equations are solved by a numerical method (i.e. gradient methods, multiple shooting, collocation). Direct methods first discretizes some variables (sequential approach, semi–infinite programming) or all variables (simultaneous approach) and solve then the resulting optimization problem numerically. Using the sequential approach one discretizes the control parameters (in our context the model error functions), determines an approximate for the controls and solves the model equations for the states exactly using the current approximate for the controls. This is iterated until the required accuracy is reached. The approach is also called control vector parameterization or, since the dynamic constraints are always fulfilled, feasible path method. The simultaneous simulation discretizes all involved variables. During the optimization procedure the constraints (here the discretized differential equations) can be violated. Therefore the method is classified as an infeasible path method. It is also denoted as the transcriptor method. At the solution, however, the algorithm has to ensure that the constraints are satisfied. For a more detailed discussion, a survey and a comparison of the various methods we refer to [19]. Above classification of methods differs from the terms used for control problems with PDE's. There one speaks of first optimize then discretize or vice versa. One typically also distinguishes as above the two approaches of viewing the optimization governed only by the controls or as a problem of all involved unknowns (all in once technique). However, in contrast to the above sequential and simultaneous approach the distinction is considered in its infinite dimensional function space setting. The control governed approach uses mostly a Newton algorithm applied to the optimization problem reduced to the controls (black box approach). Whereas the all in once ansatz employs frequently the SQP–technique such that the states are approximated by the states of linearized equations [96, 100, 101, 105, 166, 175]. The simultaneous approach has analogies to the all in once approach. However the fully discretized nonlinear optimization problem (simultaneous approach) has not to be solved necessarily by a SQP–method but for e.g. penalty or barrier function methods or an SQP–filter algorithm are employed also [49, 79, 148, 153].

For our purpose we favor the application of a direct method, since the necessary optimality conditions, which have to be known for the indirect method and which can be quite cumbersome in the presence of inequality constraints, and the switching structure of the active constraints are not easy to access for the operators. Furthermore, we concentrate on a simultaneous approach. The sequential approach would solve the differential equations in each iteration step up to a certain accuracy exactly. Given the nonlinearity this does not fit into the anytime approach. In this spirit also the application of direct solvers for the arising linear algebra is ruled out. Iterative approaches are favored on all levels of the solver. In the last years sophisticated algorithms have been developed which are capable of solving dynamic optimization problems (mainly for control problems) reliably and fast, see for e.g. [3, 16, 17, 18, 34, 40, 103, 180]. However they do not fulfill the anytime requirement.

Given the requirement to provide at any instant an estimate that increases in accuracy with the utilized computation time, the approach of recursive refinement of the approximate leads to the use of a multiscale ansatz in context with nested iterations. As a consequence one has to ensure maximal exploitation of the knowledge obtained in the previous step for the current refinement on all levels of the solver, as there are for e.g. the previous approximation, the Hessian, the active set. It demands that an algorithm is constructed where all levels of the solver have to be newly addressed. Black box optimization solvers are ruled out. Summarizing we can say that there is a large variety of issues which have to be addressed for the anytime solution of state estimation.

- One part is the signal analysis considering the noisy data.
- The other part is the optimization solver on a fixed horizon: nonlinearity, inequality constraints, the iterative solution of the arising linear equations.
- Then signal analysis and optimization has to be combined together with the recursive updating. This refinement shall be adaptive to have minimal cost for a given increase in accuracy.
- And to avoid in each iteration solving from scratch, one has to ensure maximal exploitation of the knowledge obtained previously.
- Also the information obtained on one time horizon has to be transmitted to the next.
- Last but not least the issue of ill-posedness of the original problem has to be solved with regularization techniques. Here it turned out that the proposed adaptive multiscale approach can be utilized too.

In the development of such an algorithm several conceptional and technical problems had to be solved which did not occur previously. Some of them may have arisen in a hidden way also in other approaches, as for example large condition numbers in the arising linear algebra. However due to missing awareness these problems have been ignored.

Our anytime algorithm is heavily based on the discretization with wavelets for several reasons. Wavelets built a Riesz–basis of the square integrable function space and are organized in a multiscale way. Roughly speaking they can distinguish locally the frequencies of functions. This property is one reason, that in signal analysis wavelets are a well established, efficient and widely used technique [44, 48, 129, 133, 150]. Transmission time from the signal processing to the optimization problem is minimized when a unified framework is used. This suggests the use of wavelets also for the optimization part. Moreover, due to their hierarchical organization a change of basis after refinement is avoided. The distinction of local frequencies promise in addition that the adaptive refinement based on wavelets is very efficient. Their Riesz–basis property offers the connection between the continuous in time problem with the discretized but infinite formulation, which is then restricted to a finite dimensional problem. This connection is exploited in various parts of the algorithm. Consequently, the above distinction between first discretize and then solve the finite dimensional problem or first set up the necessary conditions and then discretize the problem is more subtle for wavelets. It depends in particular on the stage when one reduces the discretization to a finite dimensional subset of the bases. Furthermore, in other applications wavelet techniques have been successfully applied for preconditioning [47, 56, 57, 120]. They eliminated the increase of the condition number occurring with refinement. Hence, the use of wavelets was promising to obtain a mesh independent number of iterations when solving the optimization problem.

Structure of the work

The research for multiscale concepts in state estimation was part of the 'Schwerpunktprogramm: Echtzeitoptimierung großer Systeme'. The project 'process monitoring using wavelet based real-time optimization' was a joint work with Prof. Wolfgang Dahmen at the 'Institut für Geometrie und Praktische Mathematik' and Prof. Wolfgang Marquardt and Dr. Thomas Binder at the 'Lehrstuhl für Prozesstechnik' at the RWTH–Aachen. The articles which have been written in the time of this joint project are influenced by all coworkers, and hence all are coauthors. In the last two articles I continued the study of state estimation with the focus on one key subproblem. I solved this issue such that it can be considered now as completed.

All of the above mentioned prospects were addressed in various degree of depths. While the project was fundamental research the cooperation with process engineers always ensured the awareness for the applicability to real world problems. In some parts of the development of new concepts the focus was on numerical simulations and computational validation while others were mathematically more rigorous. This is reflected also in this work. The first two articles give an introduction and an overview of the formulation of the problem and of the overall algorithm with its numerical validation. While the others contain more rigorous material considering specific subproblems. Obviously not all subproblems could be solved in detail. However the applicability and the potential of the developed multiscale concept could be shown.

The work is organized as follows.

The first paper *'towards multiscale dynamic data reconciliation'*, [20] published 1998, introduces and discusses in detail the problem of state estimation and the employed problem formulation. Here the estimation problem is considered from the view of data reconciliation. The model equations that are used and their extension by error functions as well as the incorporation of inequality constraints are motivated. The estimation problem is formulated in a deterministic setting as a least square problem based on the theory of inverse problems. Then the issue of ill–posedness is shortly considered and a regularized problem formulation is derived. This formulation is the starting point of the considerations in later work. Open research questions concerning the suggested problem formulation with respect to data reconciliation are summarized. In the second part of this article the new multiscale framework is presented. The algorithm including signal analysis, optimization and iterative refinement as well as moving the receding horizon is outlined. While several of the articles include a section on wavelets, this work contains the most explicit introduction. The employed methods for signal analysis based on wavelets are presented in detail and our first view on the discretization (Petrov–Galerkin discretization of the model equations) is given. A preliminary ansatz for an adaptive refinement is also mentioned. Finally, the overall algorithm is examined on a nonlinear problem with a first implementation using mostly general purpose optimization solvers.

The second paper *'multiscale concepts for moving horizon optimization'*, [28] published 2001, specifies the above multiscale approach. All occurring issues are discussed and the results obtained during the project are summarized. Regularization using a Tikhonov type technique as well as regularization by discretization is briefly described. While the rest of the paper is based on the former, the approach using the adaptive multiscale approach is shortly summarized. In contrast to the first article, now the discretization with wavelets is considered in its infinite dimensional space. Hence wavelet discretization yields a discretized problem which is equivalent to the original formulation with the unknowns in function spaces. Nevertheless, for the numerical treatment we have to choose a finite dimensional subset. The arising finite dimensional problem coincides with the above mentioned Petrov–Galerkin ansatz. For the signal processing we referred to the previous paper. Then the optimization problem on a fixed horizon with the hierarchy of subproblems is discussed in the overall spirit of the anytime ansatz. As mentioned, all levels of the solution process have to be newly addressed and, to ensure efficiency, previously obtained information has to be exploited. The employed adaptive refinement does not only take into account the sensitivity of the cost functional and a local error estimator but has to consider also the interaction with the signals (the processed inputs). The nonlinearity is currently treated with a general purpose SQP-method. For the inequality constraints numerical experiments suggest the use of the active set method with preinitializing the active set using the previous approximate. The developed method of solving the innermost subproblem, namely the arising linear systems, is shortly described including the wavelet based preconditioning. This preconditioning bounds effectively the iteration number independently of the increase of unknowns with refinement. Finally the part of moving the time horizon is outlined. Here, the approximated solution is extrapolated and then compressed with wavelet techniques. The efficiency of a uniform discretization on one scale is numerically compared with the developed adaptive refinement approach over several horizons. It shows the enormous gain in cpu time if the intermediate information is exploited. Hence a multiscale approach seems to be a promising method for efficiency not only in the context of an anytime requirement. In the development of this anytime algorithm we encounter many new problems on each level. For example the possibly extremely large, problem inherent, condition number emerged as one key problem, even though they are bounded independently of the refinement. The open problems and research issues are summarized in the conclusions.

The paper *'on the regularization of dynamic data reconciliation problems'*, [29] published 2002, demonstrates that the adaptive multiscale concept developed in the context of an anytime algorithm can serve also as a possibility to regularize the original ill–posed problem. Regularization by discretization and also regularization by employing an iterative method is known in the literature, but is still a fairly open field of research. However, to the authors knowledge there exists only a few papers on adaptive discretization for regularization [8, 94]. This paper contributes to this very interesting and promising approach for successful regularization. The adaptive multiscale scheme can be viewed as an iterative process to find a regularization which depend on the local regularity of the exact solution. High frequencies can be captured locally in the approximate solutions while they can be interpreted as noise at other positions. Computational experiments illustrate the superiority to regularization by uniform discretization. For the tested examples the adaptive multiscale scheme yields convincing regularization results.

The next papers focus on one of the occurring key problem: the subproblem of solving the arising linear equations iteratively. Most optimization solvers employ highly sophisticated direct methods exploiting the structure of the matrices. However, in the spirit of the overall concept we wish to avoid the exact solution of a subproblem, firstly to have an increase in accuracy where we can stop when computation time is over, and secondly since the exact solution of a subproblem does not lead to higher accuracy of the estimate due to the discretization error. The development of an efficient iterative algorithm turned out to be severely difficult due to the underlying system properties. The issue of ill–conditioning does not show up apparently when a direct method is used, though the solutions may be very sensitive to rounding errors or other perturbations of the original data. Considering iterative methods the problem inherent ill–conditioning was illuminated. Here the eigenvalue distribution is an important issue not only for the accuracy of the solution but also for the efficiency. Since the finally arising linear equations reflect an optimization problem resulting after linearization and after the treatment of the inequality constraints, the iterative solver including the difficult issue of preconditioning is presented in the next papers for an optimization problem with linear constraints only.

In *'iterative algorithms for multiscale state estimation, part 1: concept'*, [26] published 2001, the discretization idea based on wavelets, the multiscale framework and the nested iteration approach is presented in detail. Here, the optimization algorithm is viewed as an indirect method. The weak formulation of the first order necessary optimality conditions are provided and a Galerkin discretization based on wavelets is applied to the resulting equations. However, this view of discretization also equals the previously mentioned wavelet discretizations. Given a fixed discretization a simplified Uzawa algorithm is developed which suits the structure of the arising linear equations. For this purpose the involved Kuhn–Tucker–matrix is reduced by several unknowns. Conceptionally the simplified Uzawa algorithm is the application of the cg–method to the system reduced to the states only. Due to the discretization of differential equations one faces (roughly speaking) with finer discretization an increase of the condition numbers. We show that wavelet based preconditioning can prevent this increase. It ensures that the number of necessary iterations on each refinement level are bounded independently of the increasing number of unknowns. Hence, asymptotically the iterative solver is more efficient as a direct solver. Stopping criteria are derived to guarantee a certain error reduction after refinement of the discretization. Given the employed wavelets the best possible convergence order is obtained in numerical experiments. Moreover it is shown that the residuals for the equations can be still large while obtaining the best possible approximation error of the state estimation problem. The above asserts: the exact solutions of the equations are irrelevant. Our nested iteration algorithm outperforms also the iterative solver applied to the finest discretized equations only. Perhaps more important in view of the anytime requirement is the fact that it provides successively improved approximations at a much earlier stage.

While in part 1 an algorithm suitable to the matrix structure is presented and the asymptotic behaviour of the condition number was considered, in *'iterative algorithms for multiscale state estimation, part 2: numerical investigation'*, [27] published 2001, the numerical performance of the method is investigated. Various Krylov–space methods appropriate to different system reductions are computationally investigated since they involve different eigenvalue distribution and the efficiency of the methods depend on several properties. Also diverse preconditioners are compared. The numerical results confirm that the proposed Uzawa method in part 1 with wavelet based, diagonal preconditioning performs best. On the other hand, it turned out that the condition numbers can still be intolerably high, even if they are bounded independently of the employed mesh (wavelet set). This problem inherent ill conditioning is preliminarily discussed. Here, observability of the state via the model equations is one issue. Normalization and particular choices of regularization parameters can improve significantly the condition number which is numerically asserted. (The choice of the regularization parameters for the issue of the condition number can be argued to be in accordance with the choice in view of the originally ill–posed problem.) Nevertheless, as supported by examples they cannot always completely overcome the problem of ill conditioning. Additional problem dependent preconditioning may be inevitable.

The next article *'state estimation without regularizing the initial data'* is very recent and, hence, unpublished [33]. It is entirely analytical. Amongst other things it aims to study the origin of the problem inherent ill conditioning. First, for the solution of the nonlinear dynamic optimization problem the necessary first order optimality condition are presented. As in the discretized, finite dimensional case in 'iterative algorithms for multiscale state estimation, part 1' it turns out that the resulting equation system can be reduced by the output functions and the model error functions corresponding to the output equation. This illuminates that the choice of regularization parameters for these model error functions can be avoided. Then, to gain more insight into the origin of problem inherent ill–conditioning of the discretized problem, the case of linear model equations neglecting inequality constraints is considered in more detail. We derive a second order differential equation for the states with mixed boundary constraints. The weak formulation corresponds to the equation system obtained by first discretizing and then reducing the discretized equations. Properties such as the dependency of condition number on the regularization parameter and the model matrices are analyzed. It is shown that for any chosen value for the regularization parameter the condition number can still be arbitrarily large depending on the underlying model equations. To the authors knowledge this effect has not been considered or seen in the community yet. Moreover, the influence of perturbed data into the estimated initial data, states and model error functions is studied in detail.

Given these results for the original problem the aim of the last included paper *'preconditioning via a Schur complement method - an application in state estimation'*, [32] accepted 2003, aims at eliminating the small eigenvalues of the discretized system, which cause the problem inherent large condition numbers. Here the Schur complement is applied for eigenvalue deflation. This is in contrast to some applications, where the Schur complement is applied iteratively to reduce the complexity of the system. We first establish the theorem, which builds the foundation of this new Schur complement approach, for symmetric indefinite systems. Then the choice of the submatrix, respectively in our context of numerical state estimation, the choice of the relevant discretization subspace is motivated. Here the multiscale approach based on wavelets provides the important connection to the non discretized problem. The chosen subspace corresponds to a nonuniform, problem adapted, coarse mesh (set of wavelet functions), which ensures a good approximation of the relevant eigenfunctions of the original problem. Moreover, one can choose a priori one subspace which serves then for all refinement levels as the subspace to construct the Schur complement. Then the Schur complements on all refinement levels of the multiscale approach are well conditioned. Numerical results show the enormous reduction of necessary iterations. *Only this combination of wavelet preconditioning and Schur complement method using the knowledge of the origin of the ill conditioning enabled us to provide an efficient iterative linear solver, which is one key ingredient of the developed anytime–multiscale algorithm.*

Changes in Notation: For easier reading I would like to mention briefly a few changes of notation which occur from one article to the other.

In the first article the signals (measurements) are denoted by $\hat{\boldsymbol{y}}$, in the paper on the regularization they are named z^δ and in all the others the variable $\boldsymbol{z}$ is used. Unfortunately, in the paper on the regularization the variable z corresponds to the states and the output functions which are otherwise called $\boldsymbol{x}$, respectively $\boldsymbol{y}$. Moreover, in contrast to the other articles here the variable y summarizes the model error function and the initial value of the state for which w and x_0 are used. The notation for the specifically employed wavelets is $\Psi^{2,2}$ or simply Ψ (piecewise linear), and $\Psi^{1,1}$ corresponds to Ψ^H (piecewise constant). However, the main overlapping in notation and terms occurs in the use of Schur complement and the relevant matrices. In the first papers the Schur complement is used to reduce the size of the discretization matrix and hence the complexity of the system, while in the last paper ('preconditioning via a Schur complement method: an application in state estimation') we employ the Schur complement to deflate the small eigenvalues of this Schur complement and hence to precondition the reduced matrix. We list the notation used in the first papers and give the corresponding notation employed in the last paper in $'\ \ '$:

$\boldsymbol{X}_\Lambda$ $= '\boldsymbol{D}_\Lambda - \boldsymbol{A}'_\Lambda$;
$\boldsymbol{\mathcal{S}}_\Lambda$ $= '\boldsymbol{X}'_\Lambda$ called Schur complement in the first papers while nothing particular in the latter;
none $= '\boldsymbol{\mathcal{S}}'_\Lambda$ called Schur complement in the last mentioned paper;
$\boldsymbol{M}^{aJ}$ etc. $= '\boldsymbol{N}'$ or $'\boldsymbol{B}'$.

Acknowledgements:

With respect to my scientific work I wish to thank in particular my coauthors. At the RWTH–Aachen I collaborated with them over nearly six years in the DFG–Schwerpunktprojekt 'Echtzeit–Optimierung großer Systeme'. Prof. Wolfgang Dahmen at the 'Institut für Geometrie und Praktische Mathematik' introduced me to multiscale methods and wavelets with their large variety of applications. The vivid and fruitful discussions with him and the very active research in his group were always inspiring. Prof. Wolfgang Marquardt at the 'Lehrstuhl für Prozesstechnik' drew my attention to the problem of state estimation, always inbedded into the greater context of process monitoring and data reconciliation. The chosen problem formulation and the relevant questions were often the topics in the open and constructive discussions. He ensured the relevance of the mathematical research for engineering and application. The daily, interdisciplinary work I mainly carried out with Dr. Thomas Binder who left the 'Lehrstuhl für Prozesstechnik' to work in industry. The stimulating debates with him clarified the different views, questions, approaches and also the different vocabularies used in the two communities, in which some expressions are used for different things and different terms describe the same thing. In these rich discussions both 'worlds' were incorporated and many times they enabled the progress in the interdisciplinary work.

Moreover, I would like to thank the DFG (German Research Foundation) for their financial support during this project and the groups involved into the afore mentioned Schwerpunktprogramm, which introduced me into the various approaches for optimization. Also I am much obliged to the mathematical institute at Regensburg university which offered me the possibility to finish this work.

Last but not least I like to thank all who enabled, supported and encouraged me during this work. In particular: my parents, who were always confident and promoted my success not only in family life but also in visible, paid work; my three brothers, who trained me at a young age to live in and to establish myself in male surroundings; my partner, who supported me tremendously; he encouraged me always to continue and he does his share of family and house work, which does not seem to be self–evident for men (although a prerequisite for women to have equal prospects at work); my children, who show me daily to look at the world including mathematics always in a new and curious way with an open mind; and all the women who never questioned but supported my opinion that it is somehow possible to combine research with female life.

Finally, I wish in particular for my daughters that it will be generally more accepted and supported to combine career and motherhood, that children will not be considered as an obstruction for a woman, but as a positive part of life of which also energy and enthusiasm for research, career and any other work can rise.

Towards multiscale dynamic data reconciliation

Thomas Binder, Luise Blank, Wolfgang Dahmen, Wolfgang Marquardt

in: R. Berber, C. Kravaris (Eds.): Nonlinear Model-based Process Control, NATO ASI Series, Kluwer Academic Pub., 623-665, 1998.

Abstract

Although reconciliation of steady-state process data is routinely applied in industrial practice, the theoretical understanding of the problem and its adequate formulation in a dynamic setting is still not mature. Existing formulation approaches are based on stochastic filters, deterministic observers or mathematical programming techniques. In this contribution, we suggest a general problem formulation of dynamic data reconciliation based on the theory of ill-posed problems and their regularizations. It results in a large-scale dynamic optimization problem which requires efficient numerical solution methods in real-time under strict limitations of computational resources. We explore a novel mathematical framework for the discretization of the dynamic optimization problem and the solution of the discretized nonlinear programming problem based on multiscale approximation. The framework attempts to integrate signal processing and optimization finally leading to a fully adaptive and highly efficient numerical treatment which always provides the best possible estimate which is attainable in the allotted time period.

1 Introduction

Process measurements never reveal adequate information about the process state. This is due to the always limited number of process quantities accessible by measurement instruments as well as to unavoidable measurement errors caused by imperfect instrumentation and signal processing. Hence, there are quantitative as well as a qualitative deficiencies which cannot be removed in an industrial environment by increased instrumentation or improved maintenance procedures due to cost and technical constraints. However, imperfect and incomplete measurement information will lead to inadequately performing or even unreliable processes. Therefore, there is a significant incentive to upgrade the information content in process measurements, since any increased understanding of the process state will enable improvement of process operations, monitoring and control. The upgraded measurement information can be directly used by operators to better manually control the process, it can be fed to some operator decision support system or to some automatic control system to lead to safer process operations of increased performance.

The measurement information can be upgraded, in principle, by means of process knowledge incorporated in some mathematical model which ideally predicts process behavior in some range of operating conditions. Such a model may either result from first principles or from a large set of previously recorded and reasonably correlated process data. Obviously, the model will never perfectly predict process behavior due to unmodeled dynamics, uncertain (and potentially time-varying) parameters, or unaccounted external disturbances. Model and measurements are incorporated in some algorithm to render an estimate of all measured process quantities as well as of all those unmeasured process quantities which are additionally required to fully describe the process behavior. The objective of these algorithms is therefore to remove any error from available measurements and to yield optimal estimates of the full process state as well as of unknown process parameters.

The problem stated has been treated extensively in various branches of mathematical and engineering science focusing on different problem formulations and objectives. The first notion of the problem in the context of chemical process operations has been given in 1961 by Kuehn and Davidson [74]. These authors formulated the problem of adjusting flow and temperature measurements on a crude oil distillation tower to satisfy steady-state material and energy balances. The problem of reconciling process measurements in order to obey steady-state material and energy balances as well as any other constraint has been subsequently called *data reconciliation*. The various problem formulations studied in recent years as well as the solution algorithms taking advantage of a specific problem structure are comprehensively summarized in a recent survey by Crowe [52]. Data reconciliation is nowadays routinely applied in industrial practice (see [188] for an industrial case study) relying on general-purpose commercial or in-house software packages.

A common assumption underlying all the established data reconciliation methods and software implementations is a stationary process. Therefore, the applicability of these methods is restricted to continuously operated processes during periods when the process is reasonably close to some steady-state. Since this assumption turns out to be very restrictive in many cases, the treatment of more general transient situations is an important extension of established technology as recently affirmed in an industrial assessment of the status of real-time process optimization [119].

So far, the reconciliation of dynamic process data by means of a dynamic process model has only received limited attention. Gertler and Almasy [86] seem to have been the first addressing the distribution of measurement error among process variables to satisfy dynamic mass and energy balances in a chemical engineering context. Explicit reference to *dynamic data reconciliation* (or the like) as an extension of the steady-state problem has been made only by a few researchers including

Almasy [7, 86], Bagajewicz [10], Bequette [154], Biegler [3, 4], Darouach [63], Edgar [115, 116, 123, 132], as well as Himmelblau [99, 113] and coworkers. Due to the close relationship between dynamic data reconciliation and simultaneous parameter and state estimation in dynamic systems, there is a large body of related work in the chemical as well as, in particular, in the control engineering literature. One may distinguish deterministicly motivated observers (e.g. [143] for a review), stochastic filtering (e.g. [85, 107], or [143] for a review) and mathematical programming based estimators (e.g. [3, 51, 106, 157], and [143] for a review). Still, the theoretical understanding of dynamic data reconciliation, adequate formulations of the estimation problem, and the development of solution algorithms for large-scale industrial problems in process operations are still in their infancy.

This contribution does not attempt to give a comprehensive survey on the state of the art of dynamic data reconciliation. Such a survey would be quite difficult to focus due to a largely missing or at least not generally accepted problem statement on the one hand and due to the large body of literature on the closely related state and parameter estimation of dynamic systems on the other hand. The latter area has been summarized recently in an excellent survey paper by Muske and Edgar [143]. Instead, we suggest first a general problem formulation of dynamic data reconciliation in Section 2 which should reflect the requirements of realistic (industrial) scenarios not being biased by available solution approaches and their underlying assumptions. Relevant research issues will be pointed out. Section 3 suggests a novel mathematical approach to dynamic data reconciliation based on multiscale approximation. Some details on the numerical solution techniques in the multiscale framework will be given in Section 4. An illustrating example is provided in Section 5 to demonstrate the potential as well as current limitations of the novel approach.

2 Problem formulation

This section attempts to provide a concise problem statement of dynamic data reconciliation. It is based on previous work (e.g. see Section 1) but does not introduce restricting assumptions potentially needed for the derivation of a mathematical framework for problem solution. This general problem formulation should serve as a reference for future work where several simplifying assumptions need to be introduced to either permit theoretical analysis or to derive efficient solution algorithms taking advantage of a particular structure apparent in a restricted problem class.

Dynamic data reconciliation is *defined* as a procedure to optimally rectify the measurements taken from a process under transient conditions as well as to optimally adjust potential uncertainties in a mathematical process model (such as parameters or unmodeled disturbances) to render cleaned measurements which are consistent with predicted process dynamics. Typically, besides removing or at least smoothing unavoidable measurement error, an estimate of the process state, the unknown parameters and some additional quantities representing model uncertainty is achieved simultaneously. The measurement errors include (i) small random fluctuations caused by the irreproducibility of the measurement device, (ii) potentially time-varying offsets or biases due to miscalibration, incorrect installation or drift of the sensor, and (iii) gross errors resulting from discrete events such as malfunctioning measurement or signal processing devices on the one hand or from process upsets on the other hand [123].

The problem complexity ranges from situations, where the full process state is measured and the model is perfect, to those situations usually occurring in realistic applications, where only a small number of process quantities is accessible by instrumentation, and where there are model uncertainties of various kinds. In the first case, only rectification of erroneous measurements must be achieved, whereas combined state, parameter and other uncertainty estimates must be provided by the reconciliation algorithm in addition in the latter case. Obviously, measurements and model must reveal a certain degree of redundancy in the latter case in particular, in order to render improvement in the information content of the available measurements feasible.

In the sequel, we first discuss a priori knowledge to be coded in a process model. Then, the estimation problem will be set up and discussed.

2.1 Process model

The process model under consideration can be associated with different types of process quantities. They include

- the measured input variables $\boldsymbol{u} \in (L_2)^{n_u}$, becoming dynamic forcing functions in the process model,
- the unmeasured inputs or time-varying parameters $\boldsymbol{\delta} \in (L_2)^{n_\delta}$, leading to unknown forcing functions in the process model,
- the time-invariant parameters $\boldsymbol{\theta} = [\boldsymbol{p}^T, \boldsymbol{\pi}^T]^T$ with $\boldsymbol{p} \in \mathbb{R}^{n_p}$ and $\boldsymbol{\pi} \in \mathbb{R}^{n_\pi}$ being the unknown and the known parameters respectively,
- the state variables $\boldsymbol{z} \in (L_2)^{n_z}$, uniquely representing the process state at future times for given independent initial conditions, inputs and parameters, as well as
- the measurements $\boldsymbol{y} \in (L_2)^{n_y}$, including all measured inputs $\boldsymbol{u}$ and some of the states $\boldsymbol{z}$.

All the process quantities involved in the mathematical model are formally viewed as being real functions of continuous time t. Obviously, this is often not true for the really measured quantities which are subsequently denoted by $\check{\cdot}$. In most practical situations the measurements are sampled at discrete times. Usually (but not necessarily) there is a constant sampling rate for every measurement. In order to allow for the most general situation we assume that l_i measurements at given instants $t_i \in [\tau - T, \tau]$ are taken for every measured process quantity in some time interval $[\tau - T, \tau]$ of length T with τ being the current point in time. The values of the measured quantities $\breve{Y}_{i,j}, j = 1, \ldots l_i, i = 1, \ldots n_y$ are catenated to the vectors $\breve{\boldsymbol{Y}}_i \in \mathbb{R}^{l_i}, i = 1, \ldots n_y$. The discrete measurements on the time interval $[\tau - T, \tau]$ form the vector $\breve{\boldsymbol{Y}} = [\breve{\boldsymbol{Y}}_1^T, \breve{\boldsymbol{Y}}_2^T, \ldots, \breve{\boldsymbol{Y}}_{n_y}^T]^T$. In contrast, $\breve{y}_i \in L_2$ denotes a real function of continuous time which suitably interpolates (or extrapolates at the boundaries) the discrete measurements $\breve{\boldsymbol{Y}}_i$ on the interval $[\tau - T, \tau]$. The interpolated (real) measurements form the vector $\breve{\boldsymbol{y}} \in (L_2)^{n_y}$ which corresponds to the measurements $\boldsymbol{y}$ as predicted by the process model. This set-up allows the consideration of multi-rate sampling or even asynchronous sampling in a simple but formally sound manner. Furthermore, as we will see later, this continuous representation permits a more efficient numerical treatment of the estimation problem.

The mathematical model of the process is assumed to be given by the set of differential-algebraic equations

$$\mathbf{0} = \boldsymbol{f}(\dot{\boldsymbol{x}}, \boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}) \tag{1}$$

and initial conditions

$$\mathbf{0} = \boldsymbol{b}(\dot{\boldsymbol{x}}_0, \boldsymbol{x}_0, \boldsymbol{u}_0, \boldsymbol{d}_0, \boldsymbol{p})\,. \tag{2}$$

Here, $\boldsymbol{f} : L_2(I)^{n_x} \times L_2(I)^{n_x} \times L_2(I)^{n_u} \times L_2(I)^{n_d} \times \mathbb{R}^{n_p} \to L_2(I)^{n_x}$, $\Omega = [\tau - T, \tau]$ represents the n_x differential-algebraic model equations whereas $\boldsymbol{b} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_d} \times \mathbb{R}^{n_p} \to \mathbb{R}^r, r \leq n_x$ refers to the set of initial conditions required to guarantee existence of the solution. In this notation the subscript 0 denotes the quantities at the left boundary of Ω, $\tau - T$.

The differential-algebraic system often has a more special structure [131]; in particular, it is of index one and of semi-explicit form in many cases; then r initial values are assigned to a subset of the states $\boldsymbol{x}$ at $t = \tau - T$ (see [177] for a more detailed treatment). The system of equations includes n_z equations for the state variables $\boldsymbol{z}$ as well as $n_x - n_z$ equations for some of the unmeasured time-varying inputs and parameters $\boldsymbol{\delta}$. Often, simple signal models, i.e. a sequence of one or more integrators, are assumed to model these quantities (i.e. [85] or [67] for an example in the context of batch reactor monitoring) when no physical explanation can be employed as for example in catalyst decay modeling [84]. The unknowns $\boldsymbol{x} \in (L_2)^{n_x}$ denote an extended state vector containing the states $\boldsymbol{z}$ and the modeled quantities in the vector $\boldsymbol{\delta}$. The remaining unmodeled quantities in the vector $\boldsymbol{\delta}$ are collected in the vector of unmodeled disturbances $\boldsymbol{d} \in (L_2)^{n_d}$. For brevity, the known parameters $\boldsymbol{\pi}$ are not shown in the equations. For convenience, we assume that suitable reference values for the unmodeled disturbances and for the parameters are included in the model equations. Hence, the quantities $\boldsymbol{d}$ and $\boldsymbol{p}$ denote the difference between the reference values and the actual values; $\boldsymbol{d}$ and $\boldsymbol{p}$ are zero if the reference coincides with the actual value.

The number of equations is typically quite large; the Jacobian is always sparse reflecting the flowsheet topology on a coarse block matrix level and the structure of the unit model in every block matrix. More general models either including discontinuities or comprising also partial differential equations are possible [131] but will not be considered here in more detail.

The equations (1), (2) denote the nominal model of the process. Since the model is never perfect, there is always a mismatch between the real process and the model which needs to be taken into account explicitly. It should be noted that part of the model uncertainty has been already incorporated in the nominal model to the extent that it is transparent to the modeler by means of the unmodeled disturbances $\boldsymbol{d}$ as well as by the unknown constant parameters $\boldsymbol{p}$. However, usually unstructured uncertainty is present in addition. The simplest model of unstructured uncertainty is an additive time-varying forcing function $\boldsymbol{w} \in (L_2)^{n_w}$, $0 \leq n_w \leq n_x$, in the model equations and an additive constant $\boldsymbol{w}_0 \in \mathbb{R}^{n_w}$ in the initial conditions:

$$
\begin{aligned}
\mathbf{0} &= \boldsymbol{f}(\dot{\boldsymbol{x}}, \boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}) + \boldsymbol{W}\boldsymbol{w}\,, &(3)\\
\mathbf{0} &= \boldsymbol{b}(\dot{\boldsymbol{x}}_0, \boldsymbol{x}_0, \boldsymbol{u}_0, \boldsymbol{d}_0, \boldsymbol{p}) + \boldsymbol{W}_0\boldsymbol{w}_0\,. &(4)
\end{aligned}
$$

The rectangular matrices $\boldsymbol{W}$ and $\boldsymbol{W}_0$ are indicator matrices containing a unit element in each column where a forcing function should be considered in a model equation and are otherwise zero. Note, that there is some arbitrariness in the uncertainty description regarding the quantities $\boldsymbol{d}$ and $\boldsymbol{w}$. However, the disturbances $\boldsymbol{d}$ may enter nonlinearly in the model due to their physical interpretation whereas the error functions $\boldsymbol{w}$ always enter linearly by assumption. Neither of the two types of forcing functions are modeled. It is the decision of the modeler, which of these uncertainty representations should be included in the model. Obviously, the number of variables in $\boldsymbol{d}$ and $\boldsymbol{w}$ should be kept at a minimum for redundancy and computational efficiency reasons.

If a dynamic model for all or some of the error terms $\boldsymbol{w}$ is preferred, it can be covered by the model structure suggested above. In this case, the set of model equations (3), (4) and state vector $\boldsymbol{x}$ is appropriately extended and the indicator matrices $\boldsymbol{W}$ and $\boldsymbol{W}_0$ are adjusted accordingly. Often, such dynamic models describing unstructured uncertainty are difficult to justify.

This model formulation generalizes previous work in dynamic data reconciliation such as that by Almasy [7], Edgar [123], or Biegler [3] and coworkers. The major difference is the way in which we are explicitly dealing with different types of uncertainties as well as with the measurement in general not covering the full process state in a general differential-algebraic setting. The measurements $\boldsymbol{y} = [\boldsymbol{y}_x^T, \boldsymbol{y}_u^T]^T$, $\boldsymbol{y}_x \in (L_2)^{n_{y_x}}$, $\boldsymbol{y}_u \in (L_2)^{n_{y_u}}$, $n_{y_u} = n_u$ and $n_y = n_{y_u} + n_{y_x}$, as predicted by the model are

$$
\begin{aligned}
\boldsymbol{y}_x &= \boldsymbol{C}\boldsymbol{x} + \boldsymbol{V}_x\boldsymbol{v}_x\,, &(5)\\
\boldsymbol{y}_u &= \boldsymbol{u} + \boldsymbol{V}_u\boldsymbol{v}_u\,. &(6)
\end{aligned}
$$

The measurement errors $\boldsymbol{v} = [\boldsymbol{v}_x^T, \boldsymbol{v}_u^T]^T$, $\boldsymbol{v}_x \in (L_2)^{n_{vx}}$, $\boldsymbol{v}_u \in (L_2)^{n_{vu}}$, $0 \leq n_{v_x} \leq n_{y_x}$ and $0 \leq n_{v_u} \leq n_{y_u}$, are selected by rectangular indicator matrices $\boldsymbol{v}_x$ and $\boldsymbol{V}_u$. In the differential-algebraic setting considered here, the rectangular matrix $\boldsymbol{C}$ is an indicator matrix selecting some of the state variables $\boldsymbol{x}$ without loss of generality. Nonlinear measurement functions as usually considered in nonlinear systems theory are here included as algebraic model equations in the vector $\boldsymbol{f}$ introducing auxiliary state variables.

If the model (3) is simplified to a system of pure ordinary differential equations, the measurement equation should be generalized to

$$\boldsymbol{y}_x = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}) + \boldsymbol{V}_x \boldsymbol{v}_x \,, \tag{7}$$

$$\boldsymbol{y}_u = \boldsymbol{u} + \boldsymbol{V}_u \boldsymbol{v}_u \,. \tag{8}$$

with $\boldsymbol{g} : L_2(I)^{n_x} \times L_2(I)^{n_u} \to L_2(I)^{n_{yx}}$. This measurement model explicitly considers uncertainties in the measured inputs as well as in the measured outputs. Both types of uncertainties are present in any practical situation and are therefore accounted for. This approach has been referred to as *error–in-variables* in the context of dynamic systems by Kim et al. [115]. Our measurement model coincides in part with the literature (i.e. [4] or [157]) but differs from the formulation used for example by Edgar and coworkers in [123, 132]. These authors define the measurement errors as the *difference between the true value of the observed process quantity and the measurement* whereas here the errors $\boldsymbol{v}$ are contaminating the *outputs predicted by the model*. We prefer this formulation since the true values of the observed variables are not accessible in practice, as they always differ from the values predicted by the model.

Besides the model equations, there is additional a-priori knowledge which should be stated and used in the estimation procedure. Following Kim et al. [115] or Robertson et al. [157], lower and upper bounds can be formulated for any of the process quantities such as the states $\boldsymbol{x}$, the inputs $\boldsymbol{u}$, the disturbances $\boldsymbol{d}$, the parameters $\boldsymbol{p}$, the measurements $\boldsymbol{y}$, and the model and measurement errors $\boldsymbol{v}$ and $\boldsymbol{w}$. These additional inequality constraints are of the type

$$s_{i,min} \leq s_i \leq s_{i,max} \,, \tag{9}$$

where s_i refers to some scalar process quantity [157]. These inequality constraints must be formulated to obtain a feasible set [143]. It should be noted that the constraints also refer to a bound on the function norm for the process quantities.

In addition to the simple bounds modeled by (9) statements on the stochastic properties of the process quantities may be included if a probabilistic interpretation of the estimation problem is pursued as done in classical filtering theory. In this case, stochastic processes with certain properties are associated with the model errors $\boldsymbol{w}$ and the measurement errors $\boldsymbol{v}$. Pragmatically, uncorrelated zero mean random variables with Gaussian probability distribution and given variance-covariance information are assumed. This assumption is crucial for reducing the analytical and computational complexity of the estimation problem; at the same time it permits an interpretation of the estimation quality and supports the choice of the estimator's tuning parameters in the linear case. In the nonlinear case, which prevails in most realistic scenarios, the advantages of using such a probabilistic interpretation largely break down. The more general situation is extremely difficult to tackle even if perfect information on the probabilistic properties of the process quantities were available. For an in-depth discussion we refer to the monographs [85] and [107] or to Robertson et al. [157] who compare stochastic filtering with mathematical programming techniques.

The general problem formulation, we are interested in, aims at avoiding any assumptions on statistical properties of the process quantities and error functions. Therefore, a deterministic rather than a stochastic setting is envisaged. This decision is based on the fact that there is no justification for simple assumptions on the probabilistic properties of the model and measurement errors in almost any real situation. Since the model errors $\boldsymbol{w}$ account for unstructured uncertainty in the process model, the signal is typically correlated in time and reveals a non-zero mean on some time interval. The signals cannot be expected to be Gaussian. Similar arguments hold for the measurement errors $\boldsymbol{v}$. In these situations the filtering approaches have shown poor performance and are extremely difficult to tune (see [143] for a summarizing discussion) aside from the loss of the theoretical foundation. We accept (at least for the moment) the loss of potentially useful information on the process statistics.

2.2 Estimation problem

For reconciliation of the measured data with the process model we may require that the degrees of freedom in the model (3)-(6) are chosen in such a way that the predicted measurements $\boldsymbol{y}$ exactly coincide with the measurements $\breve{\boldsymbol{y}}$, i.e.

$$\boldsymbol{r}(t) := \boldsymbol{y}(t) - \breve{\boldsymbol{y}}(t) = 0\,, \quad \forall t \in [0, \tau] \tag{10}$$

from the startup of the process at $t_0 = 0$ to the current time τ. These degrees of freedom are $\boldsymbol{w}, \boldsymbol{v}, \boldsymbol{d}, \boldsymbol{p}$ and those r initial values $\dot{\boldsymbol{x}}_0, \boldsymbol{x}_0$ collected in $\boldsymbol{x}'_0$ for later reference to which arbitrary values can be assigned. In the sequel, these degrees of freedom will be collected in the vector $\boldsymbol{\Delta}$ for conciseness of notation.

2.2.1 Least-squares formulation

The requirement (10) would not be appropriate since the real measurements are not reflecting the measured process quantities precisely due to contamination with measurement error. A reasonable relaxed requirement is therefore a least-squares solution to equation (10), which is given by those $\hat{\boldsymbol{\Delta}}$ minimizing a cost functional Υ according to

$$\hat{\boldsymbol{\Delta}} = argmin\ \Upsilon(\boldsymbol{\Delta}) = \|\ \boldsymbol{Q}(\boldsymbol{y} - \breve{\boldsymbol{y}})\ \| \tag{11}$$

subject to the equality constraints provided by the model (3)-(6) and a feasible set of inequality constraints of type (9) for those process quantities and error functions which can be bounded by a-priori knowledge. The weighted norm of the residual $\boldsymbol{r}$ may be defined by

$$\| \boldsymbol{Q r} \|^2_{L_2([0,\,\tau])} = \int_0^\tau \boldsymbol{r}(t)^T\, \boldsymbol{Q}^T \boldsymbol{Q}\, \boldsymbol{r}(t) dt\,. \tag{12}$$

Then, (P1) is equivalent to a weighted least squares solution of (10) with the symmetric weighting matrix $\boldsymbol{Q}^T\boldsymbol{Q}$. A more general function norm could be used instead of the L_2-norm. An advantageous choice is the Sobolev norm for integer β in L_2 as given by

$$\| \boldsymbol{r} \|^2_{H^\beta([0,\tau])} = \| \boldsymbol{r}^{(0)} \|^2_{L_2([0,\tau])} + \ldots + \| \boldsymbol{r}^{(\beta)} \|^2_{L_2([0,\tau])}\,. \tag{13}$$

Here $\boldsymbol{r}^{(\beta)}$ denotes the β-th weak derivative of the function $\boldsymbol{r}$ with respect to time and is required to be square integrable. Note that the Sobolev norm with the associated Sobolev regularity β coincides with the usual L_2-norm for the choice $\beta = 0$; employing the Sobolev norm instead of the L_2-norm in (P1) therefore generalizes the least-squares formulation.

The estimation problem (P1) is a potentially large-scale dynamic optimization problem to be solved by an appropriate technique at current time τ to provide a reconciled set of measurements $\hat{\boldsymbol{y}}$ as well as estimated $\hat{\Delta}$ after the computational delay time h at $t = \tau + h$. For on-line dynamic data reconciliation the problem must be solved repeatedly with a (not necessarily) constant sampling interval of $H \geq h$. Obviously, we would like to choose H as small as possible to get the estimate with minimal delay. The choice of H will be constrained by the algorithmic complexity and by computing resources.

This optimization problem may not be computationally tractable since the computational effort may grow unboundedly with time. On the other hand, it is also not necessary to include the measurements for $t < \tau - T_{max}$ in the reconciliation. This is due to the uncertainty in the measurements $\breve{\boldsymbol{y}}$ which dominates for a sufficiently long horizon T_{max} (much larger than the dominating process time constant) the true process measurement rendering an unfavorable signal-to-noise ratio. Hence a modified least-squares problem is formulated which reads

$$\hat{\boldsymbol{\Delta}} = argmin\, \Upsilon(\boldsymbol{\Delta}) = \| \boldsymbol{Q}(\boldsymbol{y} - \breve{\boldsymbol{y}}) \|^2_{H^\beta([\tau - T,\tau])} \tag{14}$$

subject to the same equality and inequality constraints as in problem (P1) above. In order to capture all relevant measurement information, the length of the time interval T, the horizon, should be chosen as large as computationally tractable for good estimation quality as pointed out by various authors (i.e. [123] or [157]). As above, the problem is solved repeatedly with sampling interval $H > h$. This solution approach has been previously called *moving* or *receding horizon estimation* (i.e. [144, 157]).

2.2.2 Ill-posedness of the least-squares problem

Both estimation problems belong to the class of so-called *inverse problems* since we want to *determine causes for an observed effect* [76]. Most often these inverse problems do not fulfill Hadamard's postulates of well-posedness [77]. Any *well-posed problem in the sense of Hadamard* must possess the following properties:

1. A solution exists for all admissible data.
2. The solution is unique for all admissible data.
3. The solution depends continuously on the data.

If one of these properties is not fulfilled the problem is termed *ill-posed*. In the context of dynamic data reconciliation *data* refers to the measured inputs and states. These requirements are minimalistic. Obviously, there are stronger requirements, such as the convergence of the nonmeasured states to the true states (at least in the nominal case where there is neither measurement error nor model uncertainty) which are related to observability (see e.g. [137, 138, 144, 176] for a discussion in the context of state estimation).

Obviously, the estimation problems (P1) or (P2) are not set up in a reasonable way, if they are not at least well-posed in the Hadamard sense. A rigorous analysis of the nonlinear problem is quite difficult and, to our knowledge, not available yet. However, some qualitative considerations easily reveal the ill-posedness of (P1) and (P2).

We are not concerned with existence, since we can expect that there is a minimum of the cost functional Υ in both, (P1) and (P2), if the constraints set is feasible. Uniqueness is more of a concern. Let us look at the very simple case given by

$$\begin{aligned} \dot{x} &= -ax + w\,, \quad x(0) = x_0\,, & (15)\\ y &= x + v & (16) \end{aligned}$$

with known parameter a, unknown initial condition x_0 and error terms v and w. The predicted output y is given by the linear operator equation

$$y(t) = e^{-at}x_0 + \int_0^t e^{-a(t-s)}w(s)ds + v(t)\,. \tag{17}$$

The cost functional (of problem (P1)) reads

$$\Upsilon(x_0, v, w) = \|\ (y - \breve{y})\ \|^2_{H^\beta([0,\tau])} \tag{18}$$

with y taken from (17). We cannot expect to get a unique solution for the degrees of freedom since the same best $\hat{y}$ in the least-squares sense can be established by an infinite number of solutions. Remember, that there are no restrictions on the sign of the additive terms in (17). Hence, given one solution $\{\hat{y}, \hat{x}_0, \hat{v}, \hat{w}\}$, we can easily find many other solutions by arbitrarily changing x_0 or w and compensate for this influence by an adjusted v in (17). Note that there are many more subtle possibilities to get solutions with the same $\hat{y}$.

This non-uniqueness of the least-squares solution is characteristic for *many* inverse problems. An established approach to enforce uniqueness is to define the *best-approximate solution* as a *least-squares solution* of (10) where a *suitable norm* of the degrees of freedom at the solution *takes its minimum* [76]. The best-approximate solution is therefore given by the least-squares solution with the smallest possible norm of all values of degrees of freedom which include the initial conditions as well as the model and measurement uncertainties. For linear inverse problems, the best-approximate solution is unique and closely related to the Moore-Penrose (generalized) inverse of the linear operator mapping the degrees of freedom to the measurements. An alternative related approach modifies the least-squares cost functional by a regularization term penalizing the norm of the degrees of freedom, or, more generally the norm of some differential operator applied to the degrees of freedom in order to enforce uniqueness.

So far, we did not address the third requirement for a well-posed problem which asks for a continuous dependence of the solution on the data. This property guarantees that small changes in the measured data will not lead to completely different reconciled measurements and estimated degrees of freedom. Again, we rely for the moment on the results obtained for general linear inverse problems as presented in [76] and briefly summarized below.

It can be shown that the solution to the regularized problem tends to the best approximation for error free measurements if the regularization penalty approaches zero. However, for a certain fixed level of the measurement errors the difference between the regularized solution for exact data and the regularized solution for perturbed data may assume very large values if the regularization penalty approaches zero. This difference is proportional to the measurement error level and may grow without bound depending on the spectral properties of the linear operator in the equality constraints. Hence, only with non-vanishing regularization, continuous dependence of the solution on the data can be achieved.

For given regularization parameters and fixed level of the measurement error, the estimation error can be bounded by a function of the regularization penalty and the order of magnitude of the measurement error. The estimation error approaches zero for a suitable regularization adjusted to the level of measurement error as the measurement error is reduced to zero. The estimation error depends therefore only on the level of the measurement error provided the regularization penalty can be chosen in some optimal manner.

Hence, in the linear case, regularization of the least-squares problems (P1),(P2) not only renders a unique solution but also reduces the sensitivity of the solution to the data and provides a theoretical basis to bound the estimation error if the order of magnitude of the measurement error is available. Therefore, a generalization of such an approach to nonlinear problems seems to be appropriate [77] and has already been applied in system identification problems [108].

2.2.3 Regularized estimation problem

Using the regularization approach as suggested in the theory of inverse problems [77] we could consider to (heuristically) modify the cost functional in (P1) and (P2) as follows

$$\begin{aligned}\Upsilon(\boldsymbol{\Delta}) \quad = \quad & \| \boldsymbol{Q}(\boldsymbol{y}-\breve{\boldsymbol{y}}) \|^2_{H^{\beta_y}([t_0,\tau])} + \alpha_v \| \boldsymbol{R_v v} \|^2_{H^{\beta_v}([t_0,\tau])} \\ & + \alpha_w \| \boldsymbol{R_w w} \|^2_{H^{\beta_w}([t_0,\tau])} + \alpha_d \| \boldsymbol{R_d d} \|^2_{H^{\beta_d}([t_0,\tau])} \\ & + \alpha_p \| \boldsymbol{R_p p} \|^2_{L_2} + \alpha_0 \| \boldsymbol{R}_0(\boldsymbol{x}'_0 - \boldsymbol{x}'_{0,ref}) \|^2_{L_2} \ . \end{aligned} \tag{19}$$

Here, t_0 is either 0 for (P1) or $\tau - T$ for (P2). $\boldsymbol{R}_v, \boldsymbol{R}_w, \boldsymbol{R}_d, \boldsymbol{R}_p$ and $\boldsymbol{R}_0$ are nonsingular weighting matrices whereas $\alpha_v, \alpha_w, \alpha_d, \alpha_p$, and α_0 are nonnegative constants called regularization parameters.

Let us look at this objective function in the nominal case where there are no uncertainties either in the model or in the measurements. Then, the minimization of the cost functional should lead to $\hat{\boldsymbol{v}} = \hat{\boldsymbol{w}} = \hat{\boldsymbol{d}} = \boldsymbol{0}$, $\forall t$, $\hat{\boldsymbol{p}} = \boldsymbol{0}$ and $\breve{\boldsymbol{y}} = \hat{\boldsymbol{y}}, \forall t$, if the reference values for $\boldsymbol{d}$ and $\boldsymbol{p}$ are chosen as the actual values (which is always possible in the nominal case).

We would expect the cost functional to be zero in the nominal case. This could only be achieved if we knew the correct initial values which lead to $\breve{\boldsymbol{y}}$ which is impossible. Therefore, the cost functional will take at the minimum some positive value $\hat{\Upsilon} \geq \alpha_0 \| \boldsymbol{R}_0(\hat{\boldsymbol{x}}'_0 - \boldsymbol{x}'_{0,ref}) \|^2_{L_2}$. In general we cannot expect to get correct estimates by minimizing the cost functional (19), since the minimum $\hat{\Upsilon}$ might be obtained with different $\hat{\boldsymbol{x}}'_0$ and some non-vanishing and hence incorrect solution $\hat{\boldsymbol{v}}, \hat{\boldsymbol{w}}, \hat{\boldsymbol{d}}$, or $\hat{\boldsymbol{p}}$. Based on this argument, we drop the regularization of the initial values in the cost functional and formulate the estimation problem as

$$\hat{\boldsymbol{\Delta}} = argmin \ \Upsilon(\boldsymbol{\Delta}) \tag{20}$$

with the cost functional

$$\begin{aligned}\Upsilon(\boldsymbol{\Delta}) \quad = \quad & \| \boldsymbol{Q}(\boldsymbol{y}-\breve{\boldsymbol{y}}) \|^2_{H^{\beta_y}([t_0,\tau])} \\ & + \alpha_v \| \boldsymbol{R}_v \boldsymbol{v} \|^2_{H^{\beta_v}([t_0,\tau])} + \alpha_w \| \boldsymbol{R}_w \boldsymbol{w} \|^2_{H^{s_\beta}([t_0,\tau])} \\ & + \alpha_d \| \boldsymbol{R}_d \boldsymbol{d} \|^2_{H^{\beta_d}([t_0,\tau])} + \alpha_p \| \boldsymbol{R}_p \boldsymbol{p} \|^2_{L_2} \end{aligned} \tag{21}$$

subject to the equality constraints (3)-(6) and inequality constraints (9).

Note, that Rawlings and coworkers [144, 157] found in the context of nonlinear state estimation that penalizing the initial conditions does not guarantee convergence in the nominal case whereas convergence can be shown if there is no such penalty. The penalty in the initial conditions is used in [157] to optimally reflect the past measurements in a receding horizon estimator by adjusting the weight matrix in the norm according to *linear* filtering theory. This strategy is useful for short estimation horizons whereas it is not really required if the estimation horizon is taken sufficiently large [143].

For later reference in the context of the discussion of the multiscale framework in Section 5 we formulate here a simplified version of (P3) where the model (3) is substituted by a system of ordinary differential equations and the measurement model is restricted to (7). Further, we assume that the indicator matrices $\boldsymbol{W}$ and $\boldsymbol{V}_x$ are identity matrices thus allowing the error functions to contaminate every state $\boldsymbol{x}$ and measurement $\boldsymbol{y}$. $\boldsymbol{V}_u$ is assumed to be zero which corresponds to error free measured inputs. We are neither considering unknown parameters $\boldsymbol{p}$ nor disturbances $\boldsymbol{d}$. The variables with specified bounds are catenated in the vector $\boldsymbol{s}$ with upper bounds $\boldsymbol{s}_{max}$ and lower bounds $\boldsymbol{s}_{min}$. For simplicity of notation, the regularization parameters are included in the matrices $\boldsymbol{Q}, \boldsymbol{R}_v, \boldsymbol{R}_w$. In summary, these modifications yield the simplified dynamic data reconciliation problem

$$\min \parallel \boldsymbol{Q}(\boldsymbol{y} - \breve{\boldsymbol{y}}) \parallel^2_{H^{\beta_y}([t_0,\tau])} + \parallel \boldsymbol{R}_v \boldsymbol{v} \parallel^2_{H^{\beta_v}([t_0,\tau])} + \parallel \boldsymbol{R}_w \boldsymbol{w} \parallel^2_{H^{\beta_w}([t_0,\tau])} \tag{22}$$

subject to the constraints

$$\begin{aligned} \dot{\boldsymbol{x}} - \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) - \boldsymbol{w} &= \boldsymbol{0}, \\ \boldsymbol{y} - \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}) - \boldsymbol{v} &= \boldsymbol{0}, \\ \boldsymbol{s} - \boldsymbol{s}_{max} &\leq 0, \\ \boldsymbol{s}_{min} - \boldsymbol{s} &\leq 0. \end{aligned}$$

2.2.4 Relation to previous work

Some authors (e.g. [123] or [154]) use the formulation of (P1) which without any regularization is ill-posed. Their solution does not obviously reveal amplification of measurement errors since the relatively coarse discretization of continuous time acts itself as some regularization of the estimation scheme [76]. In fact, Liebmann et al. [123] observe improved estimation quality in the context of bias estimation if they penalize the bias (corresponding to our $\boldsymbol{v}$) in the least-squares objective function, i.e. if they regularize the ill-posed problem.

The formulation suggested by Rawlings and coworkers (see [157] for example) coincides with a specialization of (P3) (e.g. their formulation does not consider our $\boldsymbol{p}$ and $\boldsymbol{d}$ variables) *if we add the constraints* (10) to (P3). Inserting this equality of real and predicted measurements in (P3) eliminates the first term in the cost functional and hence leads to the objective function used by Robertson et al. [157]. These authors therefore determine the modeled measurement errors such that the real measurements coincide exactly with the predicted measurements whereas our formulation is less restrictive at the expense of a larger number of degrees of freedom.

Finally, we want to compare our approach with the work of Michalska and Mayne [137, 138]. These authors suggest a moving horizon observer for error free measurements and a perfect model using a formulation similar to (P2). They repeatedly solve an optimization problem but do not require to reach the minimum of the cost functional. Rather, they only attempt to reduce the cost functional at each step by a given factor which assures convergence of the estimated state to the real state at a given rate as time goes to infinity under an observability condition by adjusting either the initial conditions or in addition an observer input. They motivate their approach with the limited computational resources available in real-time applications. The approach suggested here is very similar to the moving horizon observer in the nominal case if the true minimum is computed in every sampling interval.

3 Research questions

In conclusion of the preceding discussion, we want to explicitly state that there is no guarantee that the formulation of the dynamic data reconciliation problem suggested here satisfies the requirements of well-posedness nor that there is convergence of all the estimated states to the true states for the nominal model. The formulation of the dynamic data reconciliation problem (P3) is still heuristic at this point although it is a reasonable extension of (P1) or (P2) which have been studied in the literature though they are definitely ill-posed. This leads us to open research questions to be tackled before the suggested problem formulation can be safely applied to dynamic data reconciliation in practice:

1. Definition of reconcilability: reconcilability should be a property of the set of measurements and the process model alone. The discussion above shows, that we cannot avoid to include the way the estimation problem is formulated. A first suggestion is as follows:

   > A set of measurements and a given process model are called reconcilable, if there exists an algorithm which renders a unique solution for the degrees of freedom in the model for all admissible data. The solution must depend continuously on the data. In the limit of vanishing measurement error it must converge to the best-approximate solution.

2. Relationship between reconcilability and observability: Obviously, observability is a much stricter requirement than reconcilability. This system property may, however, not be of very much use in situations where there is significant error (random, systematic or gross) in the measurements.

3. Analysis of (P3) and its specializations regarding estimation quality. Here, we may distinguish convergence of the estimate to the least-squares solution for vanishing measurement error as well as convergence of the estimated degrees

of freedom to the true values at least in limiting situations. This analysis may reveal necessary modifications to the suggested problem formulation. One objective of the analysis is the minimization of the number of tuning parameters and constructive guide lines for their selection.

4. Criteria and strategies to guarantee sufficient redundancy in the data and the process model to result in useful reconciled measurements and useful estimates for the degrees of freedom (see [4] for a first treatment). The selection of a set of degrees of freedom $\mathbf{\Delta}$ of minimal cardinality must also be addressed, trading off the requirements on the measurements needed for proper identification on the one and the model quality on the other hand.

5. Methods for preprocessing the measured data in order to reduce some norm of the measurement error and to provide a reasonable interpolation of the discrete measurements. The estimation quality achievable will depend on the level of measurement error as indicated by the results on linear inverse problems. Outlier removal and denoising are important requirements in this context.

6. Highly efficient numerical solution of (P3) with $t_0 = 0$ for a given set of measurements in $[0, \tau]$. This problem would be typically solved off-line to support mechanistic modeling of some (bench-scale, laboratory-scale or industrial-scale) process or to diagnose the performance of some industrial process. There are no strict limitations on computational resources. More recent contributions on numerical solution algorithms are [5, 55, 78, 182, 171, 172, 180, 181].

7. Highly efficient numerical solution of (P3) with $t_0 = \tau - T$ in a repetitive manner at a sampling interval of H for a long horizon T which takes into account the most recent measurements becoming available in $[\tau, \tau + H]$. This problem must be solved on-line in real-time obeying strict limitations of the computational resources. The numerical solution approach must take advantage of the repetitive nature and other structural properties of the problem in order to become feasible in practical implementations. A reasonable approximation to the solution of the problem must be provided within a real-time interval of H. The quality of the approximate solution should scale with H, e.g. the solution accuracy must increase with the time available for carrying out the numerical computations. Some recent contributions addressing the solution of dynamic optimization problems on a moving horizon are [3, 5, 123, 168, 190]. It should be noted that any optimization based algorithm has to somehow compete with computationally efficient observers or filters [143].

This is a demanding list. Some of these research issues have already been addressed in previous work. Nevertheless, there are no fully satisfactory solutions available and much work remains to be done. In our work to date, we have concentrated largely on the numerical aspects (items 5, 6 and 7 of the list) as presented in the next sections.

4 Multiscale dynamic data reconciliation - the framework

The numerical solution of dynamic data reconciliation problems (P3) is typically quite challenging, in particular, if on-line applications are envisaged where a reconciled set of measurements and an estimate of the degrees of freedom must be provided in real–time ideally with minimum delay to the last available measurement. Typically, the process model restricting the minimization of the cost function is large-scale. The length of the time interval T employed in moving horizon estimation should be chosen as long as possible to improve the estimation results. Obviously, due to the increasing number of discrete measurements, the computational complexity of the estimation problem increases with T. For a given problem, algorithm and hardware, the required solution time is governed by some h which may strongly vary over time depending on the availability of the process computer and on the frequencies included in the measured (and hence) estimated quantities. Despite these variations in h, the computations must *always* be accomplished in $h \leq H$. The sampling interval H for reconciled measurements and estimated values is usually much longer than most of the sampling intervals of the measurement devices but always fixed. H is chosen such that the information demand of the operating personal or of the control system (potentially including economic set-point optimization) can be satisfied.

The majority of the available algorithms for the solution of (P3) or (P4) can be subdivided into two classes with differing solution strategies. In the so-called *sequential* (or *control-vector parameterization*) *approach* (see [1, 135, 180, 181] for an example) the degrees of freedom are approximated by a polynomial or often piecewise constant function on a fixed time mesh to result in a set of algebraic decision variables. In every iteration, the differential-algebraic equations are integrated together with the sensitivity equations in order to evaluate the objective function and to provide the required gradient information with respect to the decision variables. In contrast, in the *simultaneous approach*, the state variables as well as the degrees of freedom are discretized on a fixed time mesh approximating the dynamic optimization problem by a large-scale nonlinear programming problem (NLP) (see [3, 171] for an example). Both approaches have been used to implement moving horizon estimation algorithms with varying performance. However, due to the inherently expensive integration of the model and sensitivity equations on the one hand and the significant advances in large-scale nonlinear programming on the other, the simultaneous approach may be considered more attractive in the longer run if the (typically superior) robustness of the sequential solution approach can be met by appropriate means.

All the suggested algorithms discretize all or some process quantities on a fixed time mesh. There are only few attempts to incorporate adaptivity in the discretization scheme [125, 182] which have not been used in moving horizon algorithms to our knowledge. Always, the solution is computed in a single step. Since the computation time varies with time and can only be estimated in a very rough sense, there is by no means a guarantee that a valid solution is found within $h \leq H$. To increase the chances that this restriction is met, the user can only relax the tolerances, coarsen the grid, or simplify the process model since the available upper bound on the computing time H can typically not be increased as argued above. This approach would lead to a solution of possibly inadequate accuracy in many cases; it might be over-restrictive because the algorithm most probably will not use the full amount of available computation time to further improve the solution quality.

Multiscale theory [43, 64] seems to offer attractive alternative techniques which might push the limit of large–scale moving horizon computation forward. Such techniques have just recently been applied to moving horizon estimation and control problems [122, 169, 170]. Inspired by the adaptivity inherent to multiscale theory [56] we focus in our work on a concept which steadily refines the optimization problem starting from a coarse approximation up to arbitrary degree of detail. This concept differs from the well established single step solution approach such that the optimization problem is solved repeatedly on various discretization meshes of increasing number of decision variables and approximation quality. This simultaneous approach is based on discretizing all time-varying functions using wavelets which support refinement on various scales to resolve the global trend of the functions as well as the local detail in an appropriate manner. The suggested fully adaptive method will prompt during one cycle in the moving horizon approach a valid solution of some accuracy associated with the level of refinement at any time during the course of refinement. Moreover, the method always utilizes the available computation time H, since a solution of a coarse approximation quality on level $\ell - 1$ is always available if the time limit is reached while the computations on the current level ℓ have not been completed. In addition, preprocessing of the measured data is envisaged to be treated in an integrated manner with the solution of the optimization.

The general outline of the refinement strategy is schematically given in Figure 1. Since the measurements are sampled in discrete time an appropriate transformation into continuous time representation is necessary. As it will be seen later, this functional representation is the key for a substantially compressed representation of the measurements. Compression of the measurements is used here to avoid the substantial growth of computational complexity with an increasing number of discrete measurement samples in a different manner than explored recently in [3].

The measurements, once in continuous representation, can as well be discretized in various approximation levels with wavelet based multiscale techniques. Since the measurements are corrupted with noise, outliers and the like, they are subject to pretreatment. The objective is in general to reduce the norm of the measurement error or more precisely to eliminate noise or outliers or to interpolate missing data due to a malfunctioning measurement device. According to the theoretical considerations

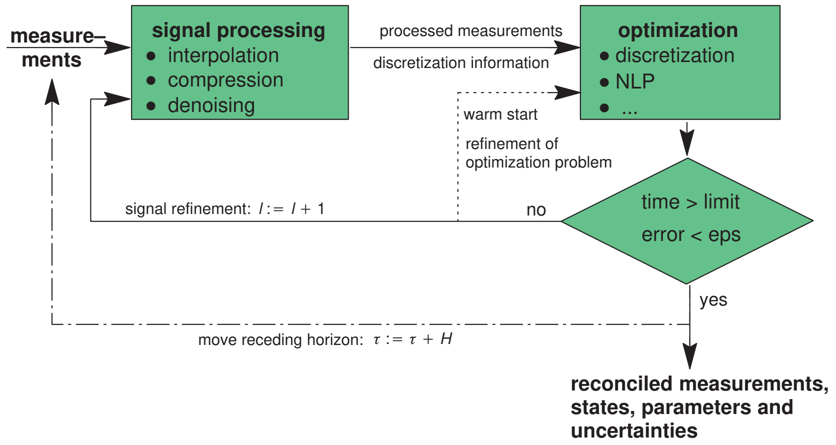


Figure 1: Schematical outline of the refinement strategy

on the solution of inverse problems in the presence of erroneous data, the reduction of some norm of the measurement error not only eases the numerical burden but also leads to smaller estimation errors caused by regularization and measurement error.

The approximated signal is then passed to the optimizer unit where the state variables and degrees of freedom of the optimization problem are discretized on some mesh related to the resolution used to represent the measurements. Thus we obtain an algebraic optimization problem which is solved with an appropriate NLP technique. The two units signal processing and optimization might appear at first glance to be separated. However, a discretization frame spanned by wavelet functions provides an integrated approach which enables easy and fast compression of the problem formulation as well as a fast data exchange between signal and optimization unit.

The numerical solution of the former refinement cycle $\ell - 1$ is reused for a warm start of the optimization in the current cycle ℓ (see [123] for the computational savings with warm start with conventional discretization) avoiding significant computing loads as compared to a start from scratch. Ideally the computational complexity in each refinement step has to be closely related to the number of the additional decision variables included in the actual refinement step. An acceptable complexity is obtained, if the solution of the last approximation can be fully utilized. This includes the evaluation of the discretized equation systems including the Jacobian and Hessian matrices as well as the numerical solution of the updated equation system. For a certain problem class the wavelet discretized equation system can be completely reused while updating. In contrast, applying Runge-Kutta or orthogonal collocation on finite elements discretizations, popular for the discretization of dynamic optimization problems on moving horizons (i.e. [3]), the discrete system of equations have to be completely reformulated and solved in every refinement step.

The adaptive mesh refinement, which includes refinement of the compressed measurement functions as well as refinement of the discretization of the states and degrees of freedom leading to a discretized system of equations of steadily increasing dimension, has to ensure a small additional computational effort compared to the overall time span H available. That way the estimation accuracy is steadily improved with small increments. The procedure is repeated until either the overall approximation error is lower than a specified tolerance or the time restrictions are violated ($h = H$). Even in the latter case where the computation of the actual approximation level is aborted, a solution of the last approximation level is still available. This way, the best valid solution achievable with given computational resources can be always prompted. Finally, the horizon window is moved on, and the multiscale dynamic data reconciliation is repeated on the next time horizon.

5 Multiscale discretization and numerical solution techniques

5.1 Introduction to wavelets

The conceptional ideas of refinement in the multiscale approach have been presented in the last section. Now the ideas have to be put on a solid mathematical basis in order to make the concept concrete. A rigorous treatment of the wavelet methodology is far beyond the scope of this paper. For a more thorough treatment, the reader is referred to [43, 56, 64]. Here we introduce only the notation and the basic concepts, which are relevant for the understanding of our approach. For simplicity, the case of scalar functions is treated, which straightforwardly extends to vector functions. Let the inner product in L_2 be defined by $\langle f, g\rangle := \langle f, g\rangle_{L_2(\Omega)} = \int_0^1 f(t)g(t)dt$ with $\Omega = [0,\ 1]$ and $f, g \in L_2$. Further, we define finite dimensional function spaces spanned by a finite number of basis functions $\varphi_{j,k}$

$$S_j := span\{\varphi_{j,k}\ : k \in \Delta_j\}$$

with the scale dependent index set Δ_j. The basis functions $\varphi_{j,k}$ are derived from a single so-called scaling function $\varphi^{(m)}$. At this point we employ pragmatically a B-spline of order m [65]. In the interior of $[0,\ 1]$ the $\varphi_{j,k}$ are obtained by dyadic dilation with powers of two and integer translation while certain special modifications are necessary near the end points of the interval. The spaces S_j are to form a multiresolution sequence

$$\ldots \subset S_j \subset S_{j+1} \subset \ldots$$

whose union $\text{clos}(\cup_j S_j) = L_2(\Omega)$ is dense. Due to the nestedness of the spaces S_j the basis $\mathbf{\Phi}_j = \{\varphi_{j,k} : k \in \Delta_j\}$ and $\mathbf{\Phi}_{j+1}$ (the set of functions will in the sequel also be interpreted as ordered array) form a two-scale relation which in compact form reads as

$$\mathbf{\Phi}_j^T = \mathbf{\Phi}_{j+1}^T \boldsymbol{M}_{j,0}\,.$$

In other words, each basis function on a coarse scale can be written as a linear combination of basis functions on the next finer scale. Since only finitely many fine scale basis functions appear independently of j in these linear combinations, $\boldsymbol{M}_{j,0}$ is a sparse matrix.

Let us assume that an arbitrary function $f \in S_{j+1}$ is approximated in some sense in S_j then some detail has to be added to get the more accurate approximation in S_{j+1}. This detail is embedded in the detail space

$$W_j := span\{\psi_{j,k} \ : k \in \nabla_j\}$$

such that W_j forms a complement space of S_j to S_{j+1}

$$S_{j+1} = S_j \oplus W_j \, .$$

The basis functions in $\boldsymbol{\Psi}_j = \{\psi_{j,k} : k \in \nabla_j\}$ of the complement space W_j are called wavelets. Since $W_j \subset S_{j+1}$ the wavelet basis on scale j can also be expressed as a linear combination by scaling functions in $\boldsymbol{\Phi}_{j+1}$ and hence

$$\boldsymbol{\Psi}_j^T = \boldsymbol{\Phi}_{j+1}^T \boldsymbol{M}_{j,1}$$

with $\boldsymbol{M}_{j,1}$ a sparse matrix. Repetition of the decomposition splits the space S_J into a sum of complement spaces

$$S_J = S_{j_0} \oplus W_{j_0} \oplus W_{j_0+1} \oplus \ldots \oplus W_{J-1}$$

where j_0 is some fixed coarsest scale and J an arbitrary but fixed finest scale. A thorough analysis [56] reveals that not all splittings are equally well suited. In fact, the complement spaces should be chosen in such a way that there exists a dual multiresolution of spaces

$$\tilde{S}_j := span\{\tilde{\varphi}_{j,k} \ : k \in \tilde{\Delta}_j\}$$

with corresponding complements splittings

$$\tilde{W}_j := span\{\tilde{\psi}_{j,k} \ : k \in \tilde{\nabla}_j\} \, .$$

They are constructed in a similar way based on a scaling function $\tilde{\varphi}^{(m,\tilde{m})}$ satisfying

$$\langle \varphi(\cdot)^{(m)}, \tilde{\varphi}^{(m,\tilde{m})}(\cdot - k) \rangle = \delta_{0,k}$$

for integer $k \in \mathbb{Z}$. Here and in the sequel, the notation $f(\cdot)$ defines a function with generalized argument $\cdot$. Moreover, the dual function spaces should possess also two-scale relations of the above form [59, 58]. For the sake of brevity we do not explicitly give the definitions for the dual function relations but will always employ a $\tilde{\cdot}$ to denote corresponding relations in the dual setting. The dual and primal single- and multiscale spaces are related through the biorthogonality property

$$\langle \boldsymbol{\Psi_j}, \tilde{\boldsymbol{\Psi}}_{\boldsymbol{j}} \rangle = \boldsymbol{I} \, , \quad \langle \boldsymbol{\Phi}_j, \tilde{\boldsymbol{\Phi}}_j \rangle = \boldsymbol{I} \, , \quad j \in \mathbb{N} \tag{23}$$

and

$$\tilde{W}_j \perp S_j\,, \quad W_j \perp \tilde{S}_j \tag{24}$$

where $\boldsymbol{I}$ denotes the identity matrix. The inner products of the functions collected in vectors are defined here and in the sequel by their inner products associated with the corresponding index set and hence form a matrix, e.g. $\langle \boldsymbol{\Psi}, \tilde{\boldsymbol{\Psi}} \rangle := (\langle \psi_{j,k}, \tilde{\psi}_{l,i} \rangle)_{(j,k)\in\nabla,\,(l,i)\in\tilde{\nabla}}$. One can show that these biorthogonality properties form essential preconditions for important stability properties of the basis, sometimes referred to a Riesz basis, explained below.

Every function $f \in L_2$ has a unique expansion

$$f = \sum_{k\in\Delta_{j_0}} \langle f, \tilde{\varphi}_{j_0,k} \rangle \varphi_{j_0,k} + \sum_{j=j_0}^{\infty} \sum_{k=\nabla_j} \langle f, \tilde{\psi}_{j,k} \rangle \psi_{j,k}\,. \tag{25}$$

Upon subsuming all basis functions $\varphi_{j_0,k}, \psi_{j,k}$ in an infinite array $\boldsymbol{\Psi}$ and likewise, all expansion coefficients $\langle f, \tilde{\varphi}_{j_0,k} \rangle, \langle f, \tilde{\psi}_{j,k} \rangle$ in a corresponding array $\boldsymbol{d} := \langle f, \tilde{\boldsymbol{\Psi}} \rangle$, (25) can be written in a compact form as $f = \boldsymbol{d}^T \boldsymbol{\Psi}$. Moreover, in addition to the uniqueness of such expansions a Riesz basis has the remarkable property that the (Euclidean) norm of the discrete array $\boldsymbol{d}$ that characterize the continuous objects $f \in L_2$ behaves essentially as the function norm, i.e.

$$\| f \|_{L_2}^2 \sim \| \boldsymbol{d} \|_{l_2}^2 = \sum_{j,k} |\langle f, \tilde{\psi}_{j,k} \rangle|^2\,. \tag{26}$$

Thus small variations in the coefficients result in small variations of the function and vice versa – a crucial property for numerical calculations. The basis constructed in [59] enjoys all these properties for any desired choices of m and $\tilde{m}$, $\tilde{m} \geq m$, $m + \tilde{m}$ even. These properties remain trivially valid for finite truncations $\Psi^J = \Phi_{j_0} \cup_{j=j_0}^{J-1} \Psi_j$ of Ψ. Thus every function $f \in S_J$ has two equally stable representations

$$\begin{aligned} f &= \sum_{k\in\Delta_J} \langle f, \tilde{\varphi}_{J,k} \rangle \varphi_{J,k} =: \sum_{k\in\Delta_J} c_{J,k}\varphi_{J,k} \\ &= \sum_{j=j_0-1}^{J-1} \sum_{k=\nabla_j} \langle f, \tilde{\psi}_{j,k} \rangle \psi_{j,k} =: \sum_{j=j_0-1}^{J-1} \sum_{k=\nabla_j} d_{j,k}\psi_{j,k} \end{aligned}$$

with $\psi_{j_0-1,1} := \varphi_{j_0,k}$ and $\nabla_{j_0-1} := \Delta_{j_0}$. To keep the notation simple we express the above as the single- and multiscale representation of f in vector form according to

$$f(t) = \boldsymbol{c}^T \boldsymbol{\Phi}_J = \boldsymbol{d}^T \boldsymbol{\Psi}^J\,. \tag{27}$$

Both representations are related by the fast wavelet transformation (FWT) $\boldsymbol{c} = \boldsymbol{T}_J \boldsymbol{d}$ where $\boldsymbol{T}_J : \boldsymbol{d} \to \boldsymbol{c}$ is typically executed in cascadic fashion by subsequent applications of the two-scale matrices $\boldsymbol{M}_{j,0}, \boldsymbol{M}_{j,1}$ (see [59, 64]). When, as in the case referred to here, these matrices are sparse, it is not hard to verify that the execution of $\boldsymbol{T}_J$ requires $O(N_J)$ operations where N_J is the cardinality of the index set Δ_J and therefore denotes the dimension of the space S_J.

Let us now comment briefly on the role of the integers $m, \tilde{m}$ appearing in the above description of the basis. The m says that the spaces S_J are exact of order m, i.e. all polynomials up to degree $m-1$ are represented exactly as

$$x^r = \sum_{k\in\Delta_j} \langle (\cdot)^r, \tilde{\varphi}_{j,k}\rangle \varphi_{j,k}\,, \quad r = 0,\ldots,m-1\,. \tag{28}$$

It is well known that polynomial exactness combined with the local supports of the $\varphi_{j,k}$ imply that smooth functions can be approximated with the order of 2^{-mj} by functions in S_j. The $\tilde{m}$ means that the dual multiresolution spaces S_j are exact of order $\tilde{m}$. On account of the biorthogonality relation (23) this in turn is equivalent to saying that

$$\begin{aligned} \int x^r \psi_{j,k}(x)dx &= 0\,, \qquad r = 0,\ldots,\tilde{m}-1\,, \\ \int x^r \tilde{\psi}_{j,k}(x)dx &= 0\,, \qquad r = 0,\ldots,m-1\,, \end{aligned} \tag{29}$$

which are often referred to as moment conditions. An immediate consequence of (29) is that one can subtract any polynomial from a function without changing its wavelet coefficients, i.e. $\langle f, \tilde{\psi}_{j,k}\rangle = \langle f-p, \tilde{\psi}_{j,k}\rangle$ where p is any polynomial of order m. Specifically, choosing p as a Taylor polynomial of order m in the support of $\tilde{\psi}_{j,k}$ shows that $\|\langle f, \tilde{\psi}_{j,k}\rangle\|$ will be quite small when f is smooth on the support of $\tilde{\psi}_{j.k}$. Consequently by far most of the wavelet coefficients in the expansion (24) are rather small. Moreover, on account of the norm equivalence (26), discarding small coefficients causes only a small error in the function. Therefore such functions may be approximated very accurately by linear combinations of only relatively very few wavelet basis functions corresponding to the large coefficients. In fact, this number may be much smaller than the dimension of the full space S_J corresponding to the highest refinement scale among the significant coefficients. Note that working with the single scale representation in S_J corresponds to a discretization based on a uniform *mesh* refinement down to width 2^{-J}. The possibly much sparser wavelet representation determined by the significant coefficients of the approximated object means to employ an individually adapted trial space which could be much smaller than S_J and thus leads to a much more economic discretization concept. Such a possibly sparse basis is denoted by

$$\boldsymbol{\Psi}_\Lambda := \{\psi_\lambda : \lambda \in \Lambda\} \tag{30}$$

where $\Lambda \subseteq \Lambda^J := \Delta_{j_0} \cup_{j=j_0}^{J-1} \nabla_j$ is an arbitrary subset of the complete wavelet basis Λ^J. It is only assumed that Λ contains the basis functions in Δ_{j_0}, i. e. the basis functions of the coarsest single scale space S_{j_0}.

By now we have established the most important ingredients in our multiscale approach and continue with the signal processing.

5.2 Signal processing

As mentioned earlier the tasks to be performed in the signal processing part are

- transformation of the discrete measurement samples into a continuous representation,
- denoising of the measurement function,
- outlier removal,
- extrapolation of missing signal parts,
- data compression, which refers to the representation of the measurement signal on various levels of resolution.

Recent developments show that wavelets are very well suited for dealing with these tasks (e.g. [11, 12, 39, 72, 124]). We restrict the following brief presentation to interpolation, denoising and compression only. For the sake of brevity we do not attempt to give a survey on the rich literature on wavelet based signal processing. Instead we illuminate only the basic principles needed to gain a more thorough understanding of the proposed approach.

The same function spaces for signal processing as well as for the discretization of the dynamic optimization problem are used. As it turns out later, this leads to an integrated approach where the processed measurement data can be passed on fast and easily to the optimization problem on various scales of resolution. To ease notation we project the time interval $[\tau - T, \tau]$ of the moving horizon onto the unity interval $\Omega = [0, 1]$ still assuming that the measurement samples are given by $\breve{\boldsymbol{Y}} \in \Omega$. Further, we consider only one measured quantity $\breve{\boldsymbol{Y}}_i$, since the discrete measurements are processed for each single quantity separately.

5.2.1 Interpolation using a multiscale basis

The discrete, noisy, possibly non-equidistant discrete measurements $\breve{\boldsymbol{Y}}_i$ can be transformed into a continuous function by a basis spanned by scaling functions. Here, an approximation based on B-splines $\varphi^{(m)}$ is used to fit the basis used later for the discretization of the dynamic optimization problem. The continuous measurement function $\breve{y}_i$ is expanded in single scale representation in S_J where J is chosen sufficiently large to consider all l_i measurement samples $\breve{\boldsymbol{Y}}_i$ such that $\breve{Y}_{i,j} = \breve{y}_i^J(t_j) = \boldsymbol{c}_{\breve{y}_i}^T \boldsymbol{\Phi}_J(t_j), \forall t_j,\ j = 1, \ldots, l_i$. This function can then be transformed into the equivalent multiscale representation $\breve{y}_i^J = \boldsymbol{d}_{\breve{y}_i}^T \boldsymbol{\Psi}^J$ applying the fast wavelet transformation.

Of course, the wavelet interpolation of the measurement function does not limit further signal cleaning to wavelet based techniques. Rather, any other classical approach may be applied. However, we expect that the use of wavelets offers a theoretically sound and unifying way of getting a reasonably preprocessed measurement signal to feed it into the optimization algorithm. It should be noted, that any preprocessing will modify the (unknown) statistics of the measurement signal.

5.2.2 Denoising

Now $\breve{y}_i^J$ is transformed into the signal $\bar{y}_i^J$ where the over bar denotes the processed interpolated signal after denoising. To accomplish this task various methods are suggested by several authors (e.g. [41, 72, 73]). Without going into detail we only mention one common method. There, the expansion coefficients $\boldsymbol{d}_{\breve{y}_i^J}$ of the interpolated signal $\breve{y}_i^J$ are replaced by modified coefficients $\boldsymbol{d}_{\bar{y}_i^J}$ according to the shrinkage algorithm

$$\boldsymbol{d}_{\bar{y}_i^J} \;:=\; \begin{cases} \boldsymbol{d}_{\breve{y}_i^J} - (\operatorname{sign} \boldsymbol{d}_{\breve{y}_i^J})\varepsilon & |\boldsymbol{d}_{\breve{y}_i^J}| \geq \varepsilon \,, \\ \mathbf{0} & |\boldsymbol{d}_{\breve{y}_i^J}| < \varepsilon \,. \end{cases}$$

This strategy interprets small wavelet coefficients as noise and reduces their magnitude by a nonlinear algorithm which leads to a smoothed (and compressed) signal. Of course, the question how to choose the threshold ε is subtle and has to be addressed with care. The more it is known about the noise and the smoothness of the signal the more rigorously ε can be determined. For example under the assumption of uncorrelated normal distributed white noise a level independent threshold ε can be derived only knowing the coefficients $\boldsymbol{d}_{\breve{y}_i^J}$. Otherwise, if little or nothing is known about the noise, ε has to be tuned in a rather heuristic manner, usually depending on location, magnitude or scale.

5.2.3 Representation of the signal in various resolutions

Keeping the refinement strategy in mind, the signal $\bar{y}_i^J$ is represented in various levels of resolutions indicated by the refinement index ℓ. Because of the norm equivalences (26), approximations of $\bar{y}_i^J$ can easily be obtained in multiscale representation retaining only those coefficients $d_{j,k},\ j,k \in \Lambda_\ell \subset \Lambda^J$ which exceed a certain threshold ε_ℓ in magnitude:

$$\bar{y}_{i,\ell} = \sum_{|d_{j,k}| \geq \varepsilon_\ell} d_{j,k}\psi_{j,k} = \boldsymbol{d}^T_{\bar{y}_{i,\ell}} \boldsymbol{\Psi}_{\Lambda_{\bar{y}_{i,\ell}}} \,.$$

The threshold ε_ℓ is chosen such that the approximation is compressed in a sense that a specified accuracy of the approximation is preserved [39, 68, 72, 124]. The signal $\bar{y}_i^J$ is first represented by a coarse approximation determined by a comparatively large threshold ε_0. The approximation accuracy is increased taking an extended set of coefficients into account which corresponds to $\varepsilon_{\ell+1} < \varepsilon_\ell$. Since the number of significant wavelet coefficients is often far less than the number of discrete measurement samples we do have a fairly compressed representation of the signal $\bar{y}_i^J$ which contributes to a reduction of the computational effort in the subsequent optimization.

Since we choose the same function spaces for signal processing and optimization the information in $\Lambda_{\bar{y}_{i,\ell}}$ can be used as an information source to select an appropriate discretization of the states and degrees of freedom in the optimization problem in a truncated multiscale basis. In addition, only the expansion coefficients $\boldsymbol{d}_{\bar{y}_{i,\ell}}$ instead of the whole sampled signal have to be exchanged and therefore fast and easy data exchange is facilitated.

5.3 Optimization problem

5.3.1 Weak problem formulation

The optimization problem (P4) should be discretized in a way that the concept of adaptivity is supported. A multiscale formulation of the process model, which permits model representation of various resolutions is obtained by a Petrov–Galerkin discretization. Following this approach the weak formulation turns out to be appropriate:

$$\min \| \boldsymbol{Q}(\boldsymbol{y} - \bar{\boldsymbol{y}}) \|^2_{H^{\beta_y}(\Omega)} + \| \boldsymbol{R}_v \boldsymbol{v} \|^2_{H^{\beta_v}(\Omega)} + \| \boldsymbol{R}_w \boldsymbol{w} \|^2_{H^{\beta_w}(\Omega)} \tag{31}$$

subject to the constraints

$$\begin{aligned}
\langle \dot{\boldsymbol{x}} - \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) - \boldsymbol{w}, \boldsymbol{\nu}_x \rangle &= \boldsymbol{0} \quad \text{for all } \boldsymbol{\nu}_x \in (L_2(\Omega))^{n_x}, & (32)\\
\langle \boldsymbol{y} - \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}) - \boldsymbol{v}, \boldsymbol{\nu}_y \rangle &= \boldsymbol{0} \quad \text{for all } \boldsymbol{\nu}_y \in (L_2(\Omega))^{n_y}, & (33)\\
\langle s_j - s_{max,j}, \delta_{t_{i,j}} \rangle &\leq 0 \quad \forall\, j = 1 \dots n_s,\ i = 1 \dots n_{s_j}, & (34)\\
\langle s_{min,j} - s_j, \delta_{t_{i,j}} \rangle &\leq 0 \quad \forall\, j = 1 \dots n_s,\ i = 1 \dots n_{s_j}. & (35)
\end{aligned}$$

Here $\delta_{t_{i,j}} = \delta(\cdot - t_{i,j})$ are Dirac distributions. $\boldsymbol{\delta}_{max}$ and $\boldsymbol{\delta}_{min}$ collect the Dirac functions in (34) and (35) respectively. They span together with $\boldsymbol{\nu}_x, \boldsymbol{\nu}_y$ the spaces of test functions. In this setting the j-th inequality constraint of the catenated vector $\boldsymbol{s}$ is forced to hold at the n_{s_j} mesh points $t_{i,j} \in \Omega, i = 1 \dots n_{s_j}$. For finite dimensional approximations of s_j, this mesh can always be chosen fine enough to guarantee the inequality constraints with sufficient accuracy. Note, that the weak formulation holds on every refinement level ℓ. Hence, the processed measurements as well as the states and degrees of freedom should be indicated by the refinement index ℓ which is omitted here and subsequently to ease notation.

To further simplify the notation we define $\boldsymbol{\mu} := (\boldsymbol{x}^T, \boldsymbol{y}^T, \boldsymbol{v}^T, \boldsymbol{w}^T)^T \in U$ and $\boldsymbol{\nu} := (\boldsymbol{\nu}_x^T, \boldsymbol{\nu_y}^T, \boldsymbol{\delta}_{max}^T \boldsymbol{\delta}_{min}^T)^T \in M$, where U and M are the product spaces $U := U^x \times U^y \times U^v \times U^w$ and $M := M^x \times M^y \times M^{s_{max}} \times M^{s_{min}}$.

Before a discrete problem can be formulated the trial spaces U and test spaces M have to be chosen properly. This choice can take advantage of the richness of biorthogonal multiscale function spaces. The best choice of a suitable wavelet basis and its dual would lead to discretized equations with very sparse or ideally (almost) diagonal Jacobian matrices to ease the subsequent numerical solution step and at the same time showing good approximation properties in signal processing as well as in optimization. At this point we choose basis functions from the B-spline family and its dual as already employed for signal processing for both U and M and explicitly state that our current choice is one of many other (probably more advantageous) options. Further work has to be done to find the best suitable basis.

The hat function $\varphi^{(2)} \in C^1$ and the corresponding wavelet functions $\psi^{(2,2)}$ are selected for the discretization of $\boldsymbol{x}, \boldsymbol{v}$, and $\boldsymbol{y}$ taking advantage of their smoothness and the interpolation property in the mesh points 2^{-j}. Further, in order to obtain a simple discretization structure $\boldsymbol{w}$ is discretized with Haar wavelets $\psi^{(1,1)}$ which are constructed from $\varphi^{(1)} \in C^0$. With this (first and therefore preliminary) choice the trial spaces have the form

$$\begin{aligned} U_\Lambda &:= U^x_{\Lambda_x} \times U^y_{\Lambda_y} \times U^v_{\Lambda_v} \times U^w_{\Lambda_w} \\ &= span\{\psi^{(2,2)}_{j,k} : j,k \in \Lambda_x\} \times span\{\psi^{(2,2)}_{j,k} : j,k \in \Lambda_y\} \times \\ &\quad\; span\{\psi^{(2,2)}_{j,k} : j,k \in \Lambda_v\} \times span\{\psi^{(1,1)}_{j,k} : j,k \in \Lambda_w\} \end{aligned}$$

where the finite index sets $\Lambda_x, \Lambda_y, \Lambda_v$ and Λ_w can be chosen adaptively in a specific manner for every refinement level ℓ. As mentioned already before, we only assume that the index set includes all scaling functions $\psi_{j_0-1,k}$ and selected functions on higher scales of the multiscale expansion. At this point, it is open how to select the basis functions in the index sets in order to achieve a good approximation.

5.3.2 Projection

As it will be seen during the course of this section the flexibility offered by a biorthogonal wavelet basis can be fully utilized to obtain a suitable discretization structure. To ease the notation and to avoid an explosion of the indices, we only consider a scalar component of the vector functions $\boldsymbol{x}, \boldsymbol{y}$ etc.. The extension to the vector case is straightforward.

First, we start with an inequality constraint (34). The trial spaces, defined in the last section, together with the test space $M^{s_{max,j}}_\Lambda = span\{\delta_{t_{i,j}},\ i = 1, \dots, n_{s_j}\}$ yield

$$\boldsymbol{d}^T_{s_j} \boldsymbol{\Psi}^{(2,2)}_{\Lambda_{s_j}}(t_{i,j}) - s_{min,j}(t_{i,j}) \quad \leq \quad 0,\ i = 1, \dots, n_{s_j}\,. \tag{36}$$

Note, that the choice of the mesh $t_{i,j}, i = 1, \dots, n_{s_j}$ is independent of the measurement samples and reflects the locations where the inequality is exactly satisfied. The inequality constraints (35) are projected in a similar manner.

In order to minimize computational work the test space M^y_Λ of the algebraic model equation (33) is chosen such that the discretization matrix for y will be the identity matrix $\boldsymbol{I}$. At this point biorthogonality of the wavelet spaces (23) comes into play. Recall that the approximation space U^y_Λ is spanned by the wavelet basis $\{\psi^{(2,2)}_{j,k} : j,k \in \Lambda_y\}$ and that there exist biorthogonal functions $\tilde{\psi}^{(2,2)}_{j,k}$ such that

$\langle \psi_{j,k}^{(2,2)}, \tilde{\psi}_{i,l}^{(2,2)} \rangle = \delta_{i,j}\delta_{k,l}$. Consequently, if $M_\Lambda^y = span\{\tilde{\psi}_{i,l}^{(2,2)} : i,l \in \Lambda_y\}$ the algebraic model equation (33) is equivalent to

$$\boldsymbol{d}_y - \langle g(x_\Lambda, u_\Lambda), \tilde{\boldsymbol{\Psi}}_{\Lambda_y} \rangle - \boldsymbol{V}_\Lambda \boldsymbol{d}_v \;=\; \boldsymbol{0}\,, \tag{37}$$

where $\boldsymbol{V}_\Lambda = \boldsymbol{I}$ for $\Lambda_y = \Lambda_v$.

The determination of the test space M_Λ^x for the differential equations (32) requires some care. For stability reasons the test space M_Λ^x has to capture the space spanned by $\dot{\Psi}_{\Lambda_x}$, i.e. the trial space of the differential operator in (32). Utilizing a Petrov-Galerkin approach we choose $M_\Lambda^x := span\{\psi_{j,k}^{(1,1)} : j,k \in \Lambda_x'\}$ as trial space such that $span\{\psi_{j,k}^{(1,1)} : j,k \in \Lambda_x'\} = span\{\dot{\psi}_{j,k}^{(2,2)} j,k \in \Lambda_x\}$ where Λ_x' denotes an adapted index set of the trial functions. This choice yields

$$\boldsymbol{X}_\Lambda \boldsymbol{d}_x - \boldsymbol{W}_\Lambda \boldsymbol{d}_w - \langle f(x_\Lambda, u_\Lambda), \boldsymbol{\Psi}_{\Lambda_x'}^{(1,1)} \rangle \;=\; 0 \tag{38}$$

with the matrices $\boldsymbol{X}_\Lambda = \langle \dot{\boldsymbol{\Psi}}_{\Lambda_x}^{(2,2)}, \boldsymbol{\Psi}_{\Lambda_x'}^{(1,1)} \rangle$ and $\boldsymbol{W}_\Lambda = \langle \boldsymbol{\Psi}_{\Lambda_w}^{(1,1)}, \boldsymbol{\Psi}_{\Lambda_x'}^{(1,1)} \rangle$. In summary, the test spaces are

$$\begin{aligned} M_\Lambda &= M_\Lambda^x \times M_\Lambda^y \times M_\Lambda^{s_{max}} \times M_\Lambda^{s_{min}} \\ &= span\{\psi_{j,k}^{(1,1)} : j,k \in \Lambda_x'\} \times span\{\tilde{\psi}_{j,k}^{(2,2)} : j,k \in \Lambda_y\} \times \\ &\quad span\{\delta_{t_{i,j}} : j = 1,\ldots,n_s,\; i = 1,\ldots,n_{s_j}\} \times \\ &\quad span\{\delta_{t_{i,j}} : j = 1,\ldots,n_s,\; i = 1,\ldots,n_{s_j}\} \end{aligned}$$

supporting a stable and adaptive discretization scheme.

5.3.3 Discretization of the cost functional

For the actual computations, the cost functional in (P4) has to be substituted by a discrete norm which can be computed utilizing only the wavelet coefficients. Straightforward substitution of the trial functions into (31) yields the discretized cost functional

$$\acute{e} = (\boldsymbol{d}_y - \boldsymbol{d}_{\bar{y}})^T \acute{\boldsymbol{Q}}_\Lambda (\boldsymbol{d}_y - \boldsymbol{d}_{\bar{y}}) + \boldsymbol{d}_v^T \acute{\boldsymbol{R}}_{\Lambda_v} \boldsymbol{d}_v + \boldsymbol{d}_w^T \acute{\boldsymbol{R}}_{\Lambda_w} \boldsymbol{d}_w \tag{39}$$

with $\acute{\boldsymbol{Q}}_{\Lambda_y} = \boldsymbol{Q}^T\boldsymbol{Q} \otimes \langle \boldsymbol{\Psi}_{\Lambda_y}, \boldsymbol{\Psi}_{\Lambda_y} \rangle$ and analogously for $\acute{\boldsymbol{R}}_{\Lambda_v}, \acute{\boldsymbol{R}}_{\Lambda_v}$. For the moment the Sobolev regularity is specialized to $\beta = 0$; i.e. differentiation of the function is avoided. The matrices in (39) exhibit sparse block finger structure which is typical for wavelet discretized systems. Applying the norm equivalences (26), the block finger structure can be simplified to a block diagonal structure due to biorthogonality of the wavelets and their duals. To allow for a more general cost functional involving Sobolev regularities $\beta \geq 0$ and thus weighting smoothness properties of the functions, we use another important property of wavelet bases which extends the Riesz basis property (26) to the Sobolev space H^β. In fact, for a certain range

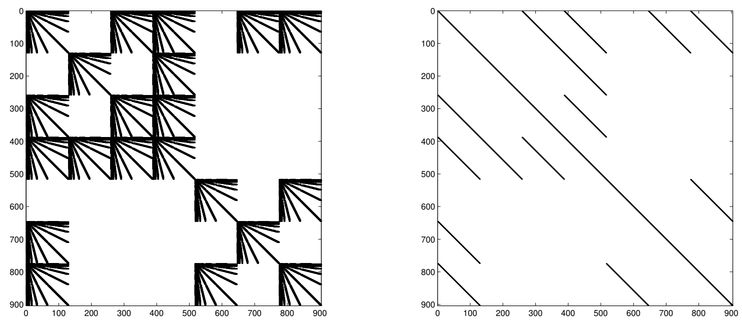


Figure 2: Structure of the discretization matrix $\acute{\boldsymbol{Q}}_{\Lambda_y}$ (left) and $\boldsymbol{Q}_{\Lambda_y}$ (right).

of regularity parameters β, depending only on the regularity of $\boldsymbol{\Psi}$ and $\tilde{\boldsymbol{\Psi}}$, one has the norm equivalence

$$\| f \|^2_{H^\beta(\Omega)} \sim \sum_{j,k} 2^{2\beta j} |\langle f, \tilde{\psi}_{j,k} \rangle|^2, \quad f \in C^\beta(\Omega)\,.$$

Thus the weighting of the smoothness of the functions in the cost functional can be easily obtained through a simple level dependent weighting of the wavelet coefficients. Therefore (39) can be replaced by

$$e = (\boldsymbol{d}_y - \boldsymbol{d}_{\bar{y}})^T \boldsymbol{Q}_\Lambda (\boldsymbol{d}_y - \boldsymbol{d}_{\bar{y}}) + \boldsymbol{d}_v^T \boldsymbol{R}_{\Lambda_v} \boldsymbol{d}_v + \boldsymbol{d}_w^T \boldsymbol{R}_{\Lambda_w} \boldsymbol{d}_w \tag{40}$$

where $\boldsymbol{Q}_{\Lambda_y} = \boldsymbol{Q} \otimes \boldsymbol{D}^\beta_{\Lambda_y}$ with $\left(\boldsymbol{D}^\beta_{\Lambda_y}\right)_{(j,k),(i,l)} = 2^{2\beta j} \delta_{j,i} \delta_{k,l}$ and analogously for $\boldsymbol{R}_{\Lambda_v}, \boldsymbol{R}_{\Lambda_w}$. Note, that $\boldsymbol{Q}_\Lambda, \boldsymbol{R}_{v_\Lambda}, \boldsymbol{R}_{w_\Lambda}$ in (40) are matrices with block diagonal structure and do not reveal the block finger structure of the direct discretization approach (39). For illustration the discretization matrices $\acute{\boldsymbol{Q}}_{\Lambda_y}$ and $\boldsymbol{Q}_{\Lambda_y}$ are shown in Figure 2 for a non-diagonal $\boldsymbol{Q}$ where the nonzero entries are represented by dots.

5.4 Solution of discretized optimization problem

The discretization employing biorthogonal multiscale approximation and test spaces results in a potentially large number of simple inequalities and a large-scale sparse nonlinear equation system (36)–(38) constraining a quadratic cost functional (40). The resulting NLP can be written in the following compact form

$$\min e(\boldsymbol{\mu}_\Lambda) \tag{41}$$

subject to the equality and inequality constraints

$$\begin{aligned} \boldsymbol{g}(\boldsymbol{\mu}_\Lambda) &= \mathbf{0}\,, \\ \boldsymbol{h}(\boldsymbol{\mu}_\Lambda) &\leq \mathbf{0}\,. \end{aligned}$$

This large-scale NLP reveals a lot of structure which must be exploited to take full advantage of the multiscale approach. At this point, only preliminary results and explorative codes have been developed to assess some of the design issues. In the following we present a straightforward application of a general purpose NLP solver and give some remarks on the research questions to be tackled when tailoring an NLP solver to the particular problem structure.

5.4.1 Solution of the NLP with a general purpose solver

In principle, (41) can be solved with any available nonlinear programming technique. However, since the function evaluations are quite expensive we are interested in solution techniques which are known to take as few iterations as possible. Here, sequential quadratic programming (SQP) or interior point SQP methods seem to be more advantageous than other alternatives like gradient or generalized Gauss-Newton methods.

Hence, we have decided to employ a general purpose SQP solver which is suitable for large-scale problems. Our current implementation employs the SQP solver RND developed by Biegler and coworkers [125, 162, 163]. Only first order information needs to be provided in addition to the residuals since the Hessian is approximated by means of the BFGS update formula. The SQP method solves the QP subproblem

$$\min[\boldsymbol{b}_\Lambda(\boldsymbol{\mu}_\Lambda^i)^T + \frac{1}{2}\Delta\boldsymbol{\mu}_\Lambda^{i\,T}\boldsymbol{\mathcal{A}}_\Lambda(\boldsymbol{\mu}_\Lambda^i)\,\Delta\boldsymbol{\mu}_\Lambda^i] \tag{42}$$

subject to

$$\begin{aligned} \boldsymbol{g}_\Lambda(\boldsymbol{\mu}_\Lambda^i) + \boldsymbol{\mathcal{G}}_\Lambda(\boldsymbol{\mu}_\Lambda^i)\Delta\boldsymbol{\mu}_\Lambda^i &= 0 \\ \boldsymbol{h}_\Lambda(\boldsymbol{\mu}_\Lambda^i) + \boldsymbol{\mathcal{H}}_\Lambda(\boldsymbol{\mu}_\Lambda^i)\Delta\boldsymbol{\mu}_\Lambda^i &\leq 0 \end{aligned}$$

in every iteration i to generate the search direction $\Delta\boldsymbol{\mu}_\Lambda^i$. Here, $\boldsymbol{\mathcal{A}}_\Lambda$ is the Hessian of the Lagrangian and $\boldsymbol{b}_\Lambda = \nabla e_\Lambda, \boldsymbol{\mathcal{G}}_\Lambda = \nabla \boldsymbol{g}_\Lambda^T, \boldsymbol{\mathcal{H}}_\Lambda = \nabla \boldsymbol{h}_\Lambda^T$ are defined as the local derivatives of the cost functional and constraints respectively. In an application of a general purpose code, the user has to provide the function evaluations $e(\boldsymbol{\mu}_\Lambda^i), \boldsymbol{g}(\boldsymbol{\mu}_\Lambda^i), \boldsymbol{h}(\boldsymbol{\mu}_\Lambda^i)$ as well as the first order information $\boldsymbol{\mathcal{G}}(\boldsymbol{\mu}_\Lambda^i), \boldsymbol{\mathcal{H}}(\boldsymbol{\mu}_\Lambda^i), \boldsymbol{b}(\boldsymbol{\mu}_\Lambda^i)$ at the current iterate $\boldsymbol{\mu}_\Lambda^i$.

In order to make this information available, inner products of wavelet functions have to be computed. For example, consider equation (38) which is included in the set of equality constraints of the NLP (41):

$$\boldsymbol{X}_\Lambda \boldsymbol{d}_x^i - \boldsymbol{W}_\Lambda \boldsymbol{d}_w^i - \langle f(x_\Lambda^i, u_\Lambda), \boldsymbol{\Psi}_{\Lambda'_x}^{(1,1)}\rangle \;\; = \;\; 0\,.$$

Note that $x_\Lambda^i = \boldsymbol{d}_x^{i\,T}\boldsymbol{\Psi}_\Lambda$, $w_\Lambda^i = \boldsymbol{d}_w^{i\,T}\boldsymbol{\Psi}_\Lambda$ are part of $\boldsymbol{\mu}_\Lambda^i$. Due to the linearity, $\boldsymbol{X}_\Lambda$ and $\boldsymbol{W}_\Lambda$ can be computed prior to the optimization run with known numerical techniques [60]. However, the evaluation of the nonlinearity is much more difficult. To date, there are no methods available which allow an efficient evaluation of this expression. The objective of current research is to find methods for the evaluation of nonlinear functions of arguments given as a wavelet expansion which avoid the complexity of the finest mesh (determined by the highest resolution in Λ, briefly referred to as full single scale dimension) but require only an effort proportional to the dimension of the adapted wavelet basis.

Pragmatically, the nonlinearity is evaluated by a method which is of the complexity of the single scale dimension at this stage of our investigations. For this purpose, the function f is evaluated pointwise first leading to a representation in terms of the single scale basis functions which are dual to the generators of $\tilde{\boldsymbol{\Psi}}^{(1,1)}_{\Lambda'_x}$. Then a fast wavelet transformation is carried out to result in

$$f({\boldsymbol{d}_x^i}^T \boldsymbol{\Psi}_{\Lambda_x}, \boldsymbol{d}_u^T \boldsymbol{\Psi}_{\Lambda_u}) \simeq {\boldsymbol{d}_f^i}^T \tilde{\boldsymbol{\Psi}}^{(1,1)}_{\Lambda'_x} \tag{43}$$

which is a multiscale approximation of f. Hence, the inner product simplifies to

$$\langle f({\boldsymbol{d}_x^i}^T \boldsymbol{\Psi}_{\Lambda_x}, \boldsymbol{d}_u^T \boldsymbol{\Psi}_{\Lambda_u}), \boldsymbol{\Psi}^{(1,1)}_{\Lambda'_x} \rangle \simeq \langle \tilde{\boldsymbol{\Psi}}^{(1,1)}_{\Lambda'_x}, \boldsymbol{\Psi}^{(1,1)}_{\Lambda'_x} \rangle \boldsymbol{d}_f^i = \boldsymbol{d}_f^i \,.$$

5.4.2 Refinement of discretization

The exactness of the solution to problem (P4) is governed by the choice of the trial space U_Λ. The key element of the proposed strategy is the successive refinement. Let Λ_ℓ denote the "old" index set which gives rise to the trial space U_{Λ_ℓ}, while the "new" trial functions, added in the refinement step, are given by the set Γ_ℓ. The optimization problem has been solved already in resolution U_{Λ_ℓ} using the SQP approach. The solution is then refined taking the additional trial functions selected by Γ_ℓ into account. Hence the resulting trial space after one refinement step is $U_{\Lambda_{\ell+1}}$ which is determined by the index set $\Lambda_{\ell+1} = \Lambda_\ell \cup \Gamma_\ell$.

The update procedure ensures that at any time the solution to the last approximated problem is available. Furthermore, this solution can also be used as an initial guess for the refined problem. Since the moment condition (29) implies decreasing wavelet coefficients with increasing scales for smooth functions, the zero augmented solution vector of resolution U_{Λ_ℓ} is a good initial guess for the refined problem (41) in resolution $U_{\Lambda_{\ell+1}}$. Hence, $\boldsymbol{\mu}^0_{\Lambda_{\ell+1}} = [\boldsymbol{\mu}^T_{\Lambda_\ell}, \mathbf{0}^T]^T$ is used as an initial guess and the changes in every entry of the old vector are expected to become smaller and smaller.

In principle, the trial functions spanning U_{Λ_ℓ} can be chosen arbitrarily. The index set Λ_ℓ might therefore contain *all* wavelet functions up to a scale J, which corresponds to an equidistant mesh. Alternatively, in an adaptive approach Λ_ℓ would consist of a strict subset of all wavelet indices up to scale J. Thus, when progressing up to higher discretization levels, the trial spaces are augmented possibly only by those subsets of the complements bases $\Psi_j, j < J$, that within some tolerance recapture most of the searched object. This provides a higher resolution for all measurements, state variables and degrees of freedom only at places where needed due to the individual behavior of these functions. The number of trial functions needed to get a certain accuracy in a nonuniform adaptive scheme is often significantly lower compared to a uniform refinement depending on the properties of the function to be approximated.

The question immediately arises how to determine Γ_ℓ in an adaptive refinement approach. Ultimately, an error estimator will be constructed on the basis of the theoretical multiscale framework such that those trial functions are selected which guarantee the best possible reduction of the approximation error. At this point, no such error estimator is available instead we employ a heuristic adaptation strategy introduced in [83]. Given the index set Λ_ℓ and the solution $\boldsymbol{\mu}_{\Lambda_\ell}$ in refinement level ℓ we search for the expansion coefficients with the largest absolute values of $\boldsymbol{\mu}_{\Lambda_\ell}$. Because of the norm equivalence property (26) these coefficients have the biggest impact on the quality of the approximation. The multiscale property together with smoothness assumptions suggest that neighbors of trial functions with large coefficients are excellent candidates for refinement. Here the wavelet functions $\psi_{j,k-1}$ and $\psi_{j,k+1}$ as well as the functions of the higher scale $\psi_{j+1,2k}$ are referred to as neighbors to the of $\psi_{j,k}$ and therefore correspond to the local neighborhood involving scale and translation. Typically not all the neighbors of the large coefficients not yet contained in Λ_ℓ are selected and added into Γ_ℓ. Rather, the additional basis functions are chosen to ensure reduction of the approximation error on the one side and a sufficiently small computation time h which is balanced against the cycle time H such that several update steps can be executed.

5.4.3 Tailored NLP algorithm

In order to establish efficiency in the algorithmical treatment the strategy for the solution of the NLP (41) presented so far is not sufficient. Rather, a tailored NLP algorithm need to be developed to take full advantage of our framework. Preliminary theoretical as well as experimental studies indicate, that there may be significant gains in efficiency if the following issues are considered during the development of a tailored NLP solver:

- exploitation of the sparse discretization structure,
- application of wavelet matrix compression techniques,
- tailored linear algebra solvers,
- efficient preconditioning of the linear systems in particular if iterative solvers are employed,
- updating of the Hessian and Jacobian, and
- the treatment of the inequality constraints.

In the following some more detailed remarks are given in order to render our current ideas a little bit more transparent.

The discretization matrices in the QP subproblem (42) exhibit a block sparse structure with sparse block matrices. The block structure reflects the structure of the cost functional and the constraints. Each block matrix reveals the particular wavelet discretization structure depending on the choice of the test and trial spaces. A proper combination may lead to more favorable patterns than those we have in our current approach. At the moment, some of the blocks of the matrices in (42) possess the wavelet typical finger structure while others are diagonal. The finger structure has $O(N \log N)$ non zero entries, where N denotes the dimension of each block. With known matrix compression techniques [61] the block finger structure can be represented by only $O(N)$ entries without significant loss of accuracy.

The underlying linear system of the QP subproblem can be solved either by factorization or by iterative techniques. Only the latter may allow for a reasonable order of complexity roughly proportional to the number of variables. Fast convergence of the iterative algorithm is a key issue. Convergence acceleration can be established through efficient preconditioning. For $J \to \infty$ diagonal preconditioners exist [36, 61] in a sense that the condition number will be independent from the number of trial functions used to discretize the system. However, more work remains to be done to develop efficient iterative solution approaches for the problem considered here.

Since we do not assume that the second order information of the model is available the Hessian is approximated using established techniques. During the course of updating the approximation to the Hessian $\boldsymbol{\mathcal{A}}_{\Lambda_{\ell+1}}$ can also be warm started augmenting the Hessian $\boldsymbol{\mathcal{A}}_{\Lambda_\ell}$ obtained from the old approximation such that positive definiteness is guaranteed. Although the elements of the Jacobian $\boldsymbol{\mathcal{G}}_{\Lambda_{\ell+1}}$ and Hessian $\boldsymbol{\mathcal{A}}_{\Lambda_{\ell+1}}$ change in each QP iteration, the parts which refer to the old trial functions given through the set Λ_ℓ stay almost constant. Therefore they only have to be updated rather than computed in each iteration step. However, this observation is based on the assumption that the mesh is already chosen fine enough such that the coarse scale components of $\boldsymbol{\mu}^i_{\Lambda_\ell}$ remain nearly unchanged when augmenting Λ_ℓ.

6 Case-studies

In this section we illustrate the principles of the multiscale approach to dynamic reconciliation using a toy example. We consider a nonisothermal continuous stirred tank reactor (CSTR) with an exothermic first order reaction. The model equations can be written in dimensionless form [178]

$$\begin{aligned} \dot{x}_1 &= -x_1 + Da(1 - x_1)\exp(\frac{x_2}{1 + x_2/\gamma}) + u_1\,, \\ \dot{x}_2 &= -(1+\beta)x_2 + BDa(1 - x_1)\exp(\frac{x_2}{1 + x_2/\gamma}) + \beta u_2\,. \end{aligned} \tag{44}$$

Here x_1, x_2 are defined as the dimensionless concentration and temperature in the reactor whereas u_1, u_2 are the associated feed concentration and cooling temperature respectively. Da is the Damkohler number, β is the dimensionless heat transfer coefficient, and B is the adiabatic temperature rise. Note, that the time in the above formulation is as well normalized. The process parameters used in this example are $(Da = 0.162, B = 14.0, \beta = 3.0, \gamma = \infty)$. In our numerical experiments, we consider only one fixed horizon of the length $[0,\ T]$ with T = 10 and project this horizon on the normalized interval $\Omega = [0,\ 1]$. We do not yet consider a moving time window. The "true" (or nominal) process state variables are simulated by integration of the model equations with $u_1 = u_2 = 0$ starting with the initial conditions $\boldsymbol{x}_0 = [0.35, 0.9]^T$. In this setting the dimensionless temperature x_2 is considered to be the measured variable. The measurements $\check{\boldsymbol{Y}}$ which are corrupted by normally distributed uncorrelated noise are sampled on an equidistant mesh with step size 2^{-J}, $J = 7$. The goal of the optimization problem is to obtain reconciled estimates $\hat{x}_1, \hat{x}_2$ of concentration and temperature from the temperature measurements $\check{\boldsymbol{Y}}$ for some $\boldsymbol{Q}, \boldsymbol{R}_v, \boldsymbol{R}_w$ utilizing the model (44).

The test and trial spaces are chosen as outlined in Section 5.3.1 and 5.3.2. In the first step, we interpolate the measurement samples $\check{\boldsymbol{Y}}$ by means of B–splines $\varphi^{(2)}$ to obtain the continuous representation $\check{y}^7$. Since this B–spline is a piecewise linear function the samples are interpolated also by a piecewise linear function. The measurement function $\check{y}^7$ then is transformed into the multiscale representation and denoised with an appropriate chosen threshold parameter ε. The nominal temperature x_2, the measurement samples $\check{\boldsymbol{Y}}$, the interpolated measurements $\check{y}^7$ and the processed measurements $\bar{y}^7$ are shown in Figure 3. Occasionally sharp spikes can be seen in the denoised signal. With the use of higher order trial functions, i.e. quadratic ($m = 3$) or cubic B–splines ($m = 4$), the representation will be smoother.

In the example considered here, the optimization problem is formulated and solved starting from a coarse approximation $\ell = 0$ with three subsequent refinement steps ($\ell = 0, \ldots, 3$) employing uniform refinement: Each quantity $\bar{y}, \boldsymbol{x}, y, v, \boldsymbol{w}$ in the optimization problem (P4) is discretized by wavelet functions of the index set $\Lambda_\ell^{J_\ell}$ with $J_\ell = 2, \ldots, 5$. This choice represents all trial functions of $S_1 \oplus W_1 \oplus \ldots \oplus W_{J_\ell - 1}$ and is equivalent to an equidistant mesh with step size 2^{-J_ℓ}. To ease notation we ommit the index J_ℓ in $\Lambda_\ell^{J_\ell}$. Figure 4 reveals the processed measurement function $\bar{y}_{\Lambda_\ell} = \boldsymbol{d}_{\bar{y}_\ell}^T \boldsymbol{\Psi}_{\Lambda_{\bar{y}_\ell}}$ which are used as input to the optimization problem. The reconciled states $\hat{x}_{i,\Lambda_\ell} = \boldsymbol{d}_{\hat{x}_{i,\ell}}^T \boldsymbol{\Psi}_{\Lambda_{\hat{x}_{i,\ell}}}$ of the optimization based on the choice of $\boldsymbol{Q} = \boldsymbol{R}_v = \boldsymbol{R}_w = \boldsymbol{I}$ are shown in the Figures 5-6. Note that starting from a very coarse approximation where each quantity has only 5 degrees of freedom the approximation accuracy is steadily refined by halving the mesh size in each refinement step.

During the course of refinement the change of the shape of the reconciled quantities is quite remarkable. By contrast, the coefficients $\boldsymbol{d}_{\hat{x}_{i,\ell}}$ in the expansion stay almost constant. The coefficients of $\boldsymbol{d}_{\hat{x}_{i,\ell}}$ which correspond to the first nine wavelet trial functions (i.e. the functions of the trial space $S_1 \oplus W_1 \oplus W_2$) are shown in Figure 7 during refinement of the approximation. Recall that in the case of uniform refinement the wavelets in Λ_ℓ, i.e. the basis function for the coarse approximation,

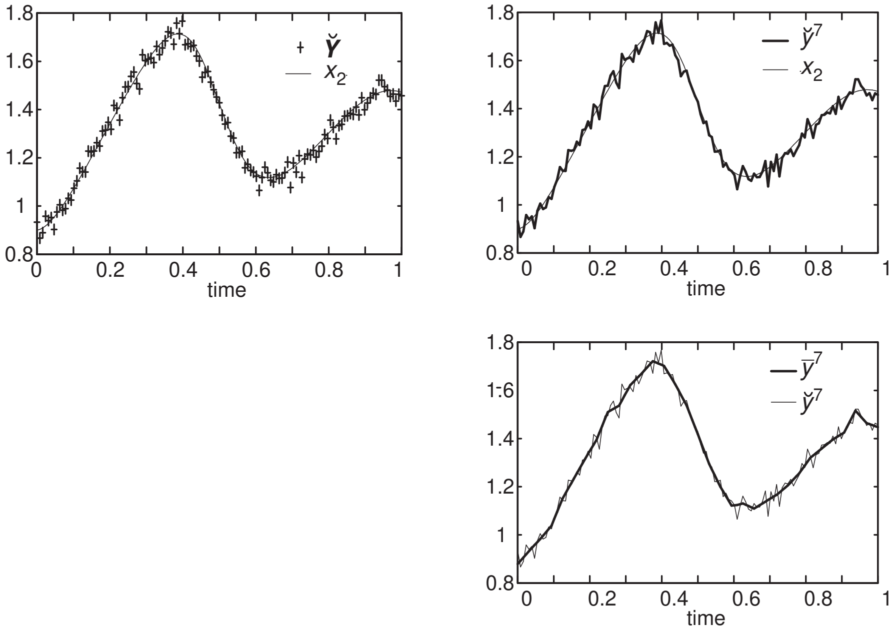


Figure 3: Measurements, original and processed measurements.

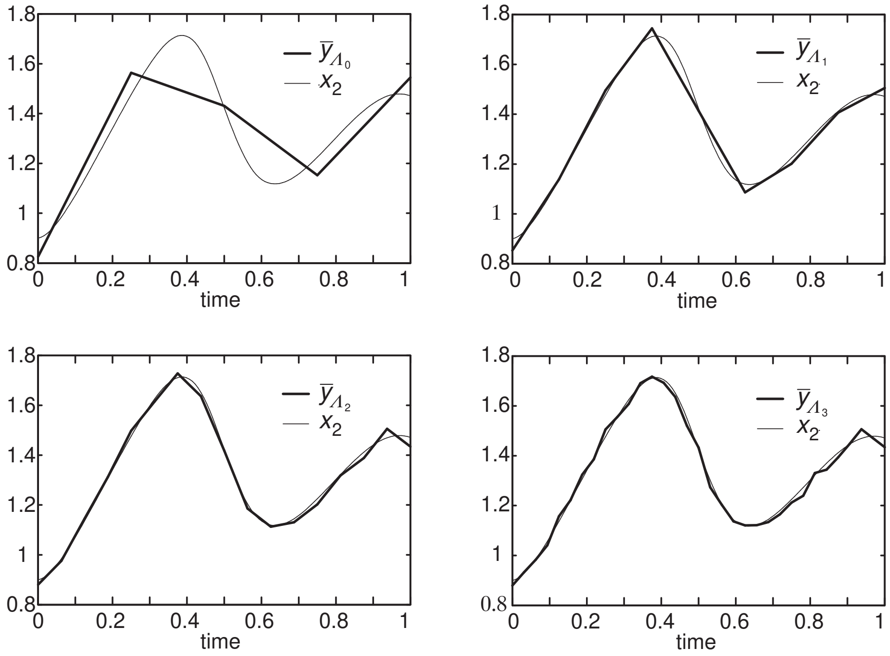


Figure 4: Processed measurement function on various scales.

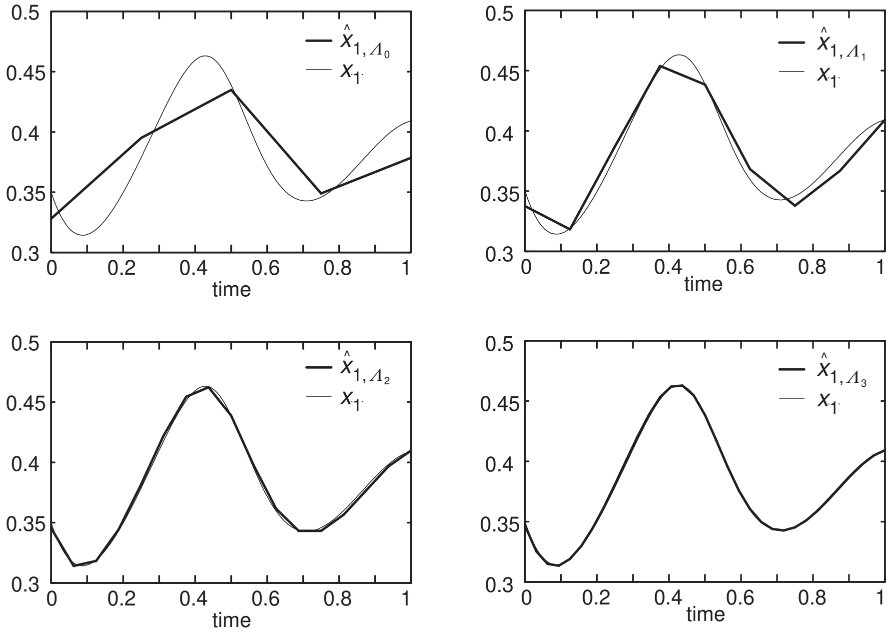


Figure 5: Refinements and original of state x_1.

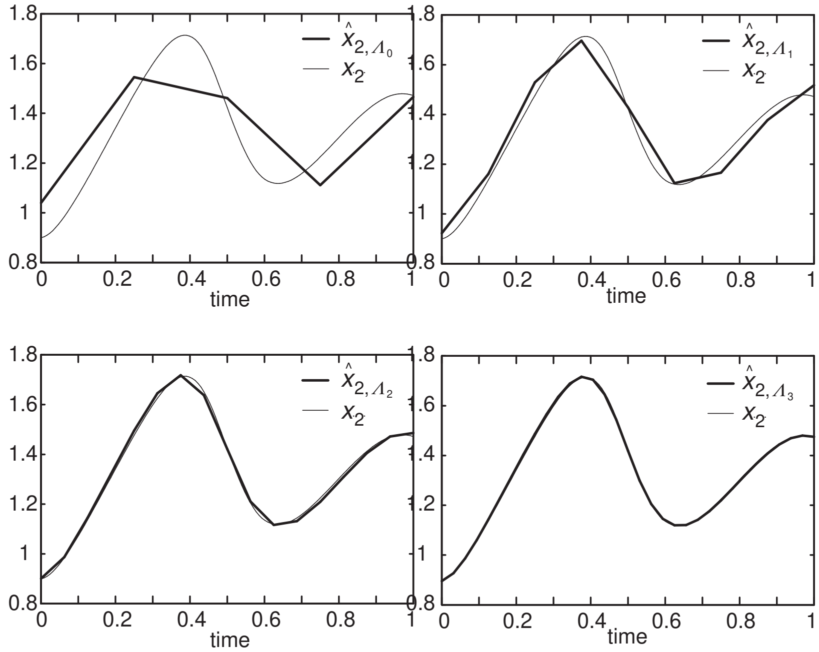


Figure 6: Refinements and original of state x_2.

are also contained in $\Lambda_{\ell+k}$, $k \geq 0$. The wavelet functions that span W_2 (the last four wavelet functions in the graph) are not contained in the initial coarse approximation ($\ell = 0$) and therefore take the value zero. As it can be seen, the changes in the coefficients are hardly noticeable. Moreover, the wavelet coefficients $\boldsymbol{d}_{\hat{x}_{i,\ell}}$ decrease in magnitude with increasing scale. As mentioned before, this is due to the moment condition and the smoothness of the approximated function. Again, levelwise refinement is only one strategy offered by the discretization scheme presented. It leads to a large number of degrees of freedom after few refinement steps. On the other hand, there is no need to increase the resolution uniformly. In this example, however, the levelwise refinement strategy is very favorable due to the nature of the approximated process quantities and a nonuniform adaptive strategy does not result in significant reduction of the effort. In other examples we have studied, a significant reduction can be achieved by nonuniform refinement as outlined in Section 5.4.2.

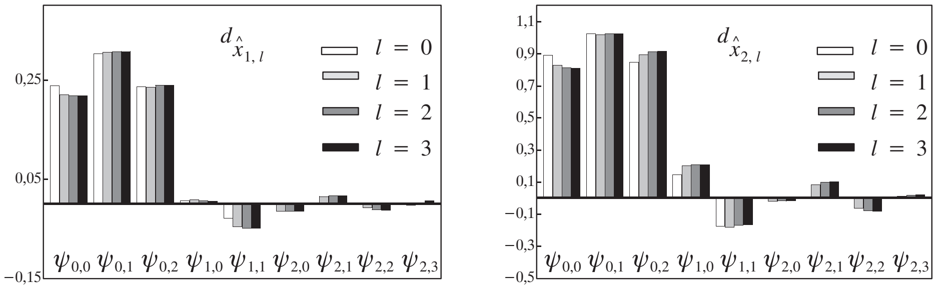


Figure 7: Wavelet coefficients during refinements of $\hat{x}_1$ and $\hat{x}_2$.

7 Conclusions

We have sketched a novel treatment of dynamic data reconciliation problems. In the first part, it suggests a purely deterministic formulation of the problem based on the theory of inverse problems. At this point, the formulation is still partly heuristic and based on theoretical evidence gained from the understanding of general linear inverse problems. A theoretical analysis of the estimation problem for the linear as well as the nonlinear data reconciliation problem is subject to future work in order to show the appropriateness of the suggested formulation. In particular, an assessment of the estimation quality is essential.

The second part suggests a new framework to the numerical treatment of dynamic data reconciliation. It is based on multiscale techniques for the discretization of the dynamic optimization problem to result in a nonlinear programming problem as well as for the solution of the underlying system of equations. The presentation is limited to an outline of the general ideas underlying the multiscale framework. The technical realization of the conceptional ideas is still at the very early stage of research. Hence, all the techniques presented at this point must be viewed as being of preliminary nature. These first results demonstrate, however, convincingly the potential of the approach to finally result in a highly efficient numerical technique which is able to provide the best possible estimate for given computational resources.

Obviously, much work remains to be done. We will concentrate on the one hand on a theoretical analysis of the linear reconciliation problem first. We are interested in bridging the gap between the theory of inverse problems and observer theory, both employing a deterministic setting for the estimation. The numerical research will focus on improved multiscale discretization schemes which can be integrated with signal processing and on the development of tailored methods for the solution of the mathematical programming problem.

ACKNOWLEDGEMENT

The authors acknowledge the contribution by A. M'hamdi. The work presented here was supported by the "Deutsche Forschungsgemeinschaft" under grant no. MA1188/6 which is also gratefully acknowledged.

Multiscale Concepts for Moving Horizon Optimization

Thomas Binder, Luise Blank, Wolfgang Dahmen, Wolfgang Marquardt

in: M. Grötschel, S.O. Krumke, J. Rambau (Eds.): Online Optimization of Large Scale Systems Springer–Verlag Berlin Heidelberg, 341–362, 2001.

Abstract

In chemical engineering complex dynamic optimization problems formulated on moving horizons have to be solved on–line. In this work, we present a multiscale approach based on wavelets where a hierarchy of successively, adaptively refined problems are constructed. They are solved in the framework of nested iteration as long as the real–time restrictions are fulfilled. To avoid repeated calculations previously gained information is extensively exploited on all levels of the solver when progressing to the next finer discretization and/or to the moved horizon. Moreover, each discrete problem has to be solved only with an accuracy comparable to the current approximation error. Hence, we suggest the use of an iterative solver also for the arising systems of linear equations. To facilitate fast data transfer the necessary signal processing of measurements and setpoint trajectories is organized in the same framework as the treatment of the optimization problems. Moreover, since the original estimation problem is potentially ill–posed we apply the multiscale approach to determine a suitable regularization without a priori knowledge of the noise level.

Key Words: model predictive control, receding horizon estimation, dynamic optimization, optimal control, on–line computation, ill–posed problem, multiscale concept, wavelets, adaptive refinement, iterative solvers.

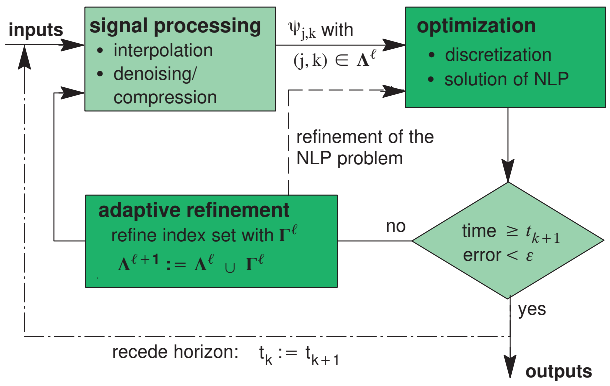
1 Introduction

The numerical solution of dynamic optimization problems is quite challenging for large–scale problems. The challenge becomes even more severe when real–time applications formulated on moving horizons such as model predictive control (MPC) or receding horizon estimation (RHE) are envisaged, since the response time where a solution has to be prompted is fixed. For a more detailed introduction into moving horizon optimization we refer to [19] in this book and references therein. Both, the regulator problem and the estimation problem have to be solved repetitively within a fixed time span ΔT which is dictated by the dynamics of the process. An acceptable algorithm has to prompt the optimal values within ΔT since proper process operation cannot be guaranteed otherwise.

The majority of known implementations of dynamic optimization algorithms for large–scale systems are based on the direct approach [19] since it does not require any analytic expression for the necessary optimality conditions which can be quite cumbersome to determine especially in the presence of inequality constraints. For this class of methods the dynamic optimization problem is transformed into a finite–dimensional nonlinear programming problem (NLP) by either discretizing the states and control profiles (simultaneous approach) [54] or by parameterizing the control variables only (sequential approach) [180]. It is common practice to approximate the time–variant quantities on a discretization mesh of fixed resolution. This applies to sequential as well as to simultaneous approaches, although in the sequential approach a stepsize adaption is used in the integrator to control the error in the state equations. The particular discretization mesh is typically chosen based on a rather conservative estimate of the computing time needed to solve the optimization problem. In recent years sophisticated algorithms have been developed which are capable to solve dynamic optimization problems reliably and fast, see e.g. [34, 40, 180], but they employ a fixed mesh and are therefore of given complexity. However, due to the nonlinear nature of the problem it is impossible to estimate exactly the necessary computation time. Hence, either a certain fraction of the available time span remains unused or the algorithm may even fail to prompt the solution in due time. Furthermore, an accurate estimate of the available computation time is hindered by competing software processes which run simultaneously on multitasking process control systems. As a consequence the available computation time is a priorly unknown and ΔT only gives an upper limit. This suggests a new view on **real time requirements** in on–line computations: *Provide an approximate solution at any time in ΔT with increasing approximation quality.*

The conceptual framework we want to present now hinges on this real time requirement. Instead of keeping the degrees of freedom fixed we propose the solution of a suitable **hierarchy of optimization problems** of increasing resolution using the simultaneous approach. We start with a very coarse approximation of states and control profiles in the optimization problem. Thus already after a hopefully very short period of time at least the coarsest approximate solution can serve as a minimal response. During the remaining time this initial solution is then **successively refined** by an **adaptive strategy** so that the full available time span ΔT is exploited in an optimal way. Moreover, for any discretization level the corresponding discrete problem has to be solved only with an accuracy that is comparable to the corresponding discretization error. Such a concept can only offer significant advantages over simply using a hierarchy of conventional discretizations if the work needed to compute an approximate solution on a coarser level need not be repeated

Figure 8: Receding Horizon Estimation.



when progressing to the next finer approximation. Hence, in the very spirit of classical *nested iteration* [35], we exploit the current approximation as initial guess and reduce the current error only by a fixed factor when progressing to the next discretization level. An extension of the refinement concept to the sequential solution approach is given in [30].

Figure 8 sketches our approach tailored to the receding horizon estimation problem. The discrete measurements are processed by denoising, compression and data fitting schemes to produce functions. This step provides also an initial mesh, respectively a first set of basis functions Λ^0, for discretizing the optimization problem. A NLP solver determines a solution of the optimization problem on the coarse initial mesh. Then, if time permits, this solution is updated in an adaptive way. The coarse mesh solution of the optimization problem as well as the potentially denoised inputs function as indicators for the new, refined mesh, respectively for the new basis index set $\Lambda^{\ell+1}$ where ℓ denotes the refinement counter. The optimization problem is resolved on this refined mesh with additional input information and by exploiting any available information from previous stages. As long as time permits this refinement procedure is repeated, until the time horizon is moved eventually after an elapsed time of ΔT. The subsequent calculations exploit again the information of the previous time horizon.

A *unified framework* for signal (input) processing and optimization is to facilitate a fast data exchange between these two conceptual blocks. We propose to realize the above concept with the aid of **wavelet methods** for the following reasons. Firstly, wavelets constitute a very efficient and established tool for denoising and compression in signal processing [41, 43, 73]. Secondly, a multiscale representation of the problem variables in the numerical solution avoids a change of bases in the updating procedure. Moreover, wavelet properties can be exploited for example for preconditioning and for adaptive refinement of the solution. For the reader who is unfamiliar with wavelets we include in Section 3 a short introduction to some of their basic properties and indicate their significance in the present context. For a more detailed treatment of wavelets we refer to [43, 46, 56, 64].

Before we proceed with the details of the refinement concept we introduce briefly a general problem formulation of receding horizon optimization problems. More details with an emphasis on state estimation can be found in [25, 24]. The dynamic behavior of the plant is often modeled by a system of differential–algebraic equations which, together with bounds on selected variables, form the constraints of the optimization. However, this work is only based on models where the state equations are described by ordinary differential equations. For a first step towards an extension to differential–algebraic equations we refer to [109]. The control functions are denoted by $\boldsymbol{u}$ and the parameters by $\boldsymbol{p}$. For *state estimation* both, $\boldsymbol{u}$ and $\boldsymbol{p}$ are given. The goal is to estimate the output functions $\boldsymbol{y}$, which correspond to the signals, and the state functions $\boldsymbol{x}$ on the receding fixed time interval $[t_{k-m}, t_k]$. Here t_k denotes the current time. The underlying measurements are typically discrete and noisy. So they have to be transformed into denoised functions $\boldsymbol{z}$. Additive model correction terms are introduced into the model equations as functions $\boldsymbol{v}$ and $\boldsymbol{w}$ which have to be estimated as well. The introduction of model correction terms actually may enhance ill–posedness as discussed in Section 2. For given $\boldsymbol{z} \in (L_2)^{n_y}$, $\boldsymbol{p} \in \mathbb{R}^{n_p}$ and $\boldsymbol{u} \in (L_2)^{n_u}$ and unknown functions $\boldsymbol{x} \in (H^1)^{n_x}$, $\boldsymbol{y} \in (L_2)^{n_y}$, $\boldsymbol{v} \in (L_2)^{n_v}$ and $\boldsymbol{w} \in (L_2)^{n_w}$ with $n_w \leq n_x$, $n_v \leq n_y$, and time–invariant indicator matrices $\boldsymbol{W}, \boldsymbol{V}$ the resulting optimization problem has the form

$$\min_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{v},\boldsymbol{w}} \int_{t_{k-m}}^{t_k} \left\{ (\boldsymbol{y}-\boldsymbol{z})^T \boldsymbol{Q} (\boldsymbol{y}-\boldsymbol{z}) \right\} \, d\tau \tag{45}$$

subject to the constraints

$$\begin{aligned}
\dot{\boldsymbol{x}} - \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u},\boldsymbol{p}) - \boldsymbol{W}\boldsymbol{w} &= \boldsymbol{0}\,, && (46)\\
\boldsymbol{y} - \boldsymbol{g}(\boldsymbol{x},\boldsymbol{u},\boldsymbol{p}) - \boldsymbol{V}\boldsymbol{v} &= \boldsymbol{0}\,, && (47)\\
\boldsymbol{l}_{\boldsymbol{x}}(t) \leq \boldsymbol{x}(t) &\leq \boldsymbol{s}_{\boldsymbol{x}}(t)\,, && (48)\\
\boldsymbol{l}_{\boldsymbol{y}}(t) \leq \boldsymbol{y}(t) &\leq \boldsymbol{s}_{\boldsymbol{y}}(t)\,, && (49)\\
t \in [t_{k-m}, t_k] & \qquad .
\end{aligned}$$

The size of the problem is dominated by the number of state functions n_x.

In the sequel, we focus on the estimation problem only, but want to emphasize the similarity to general dynamic optimization problems such as the *model predictive control* problem. There, typically the parameters $\boldsymbol{p}$ and the initial conditions $\boldsymbol{x}_0 = \boldsymbol{x}(t_k)$ are known and the controls $\boldsymbol{u}$ are to be determined, while model correction terms are not present. The signals $\boldsymbol{z}$ represent in this case the given reference trajectories to be tracked by control. A very similar optimization problem to (45)–(49) is to be solved to determine the controls $\boldsymbol{u}$.

2 Problem Regularization

The goal of the problem formulation (45)–(49) is to provide good estimates for $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{v}, \boldsymbol{w}$ from noisy measurements. If no $\boldsymbol{v}$ and no inequalities are present a minimal pre–requisite for a unique solution is a sufficient number n_y of measured model outputs. Moreover, observability of the model has to be guaranteed, i.e. given $\boldsymbol{y}, \boldsymbol{v}, \boldsymbol{w}, \boldsymbol{u}, \boldsymbol{p}$ there has to exist a unique solution $\boldsymbol{x}$ of the model equations (46)–(47). For a general discussion of necessary and sufficient conditions concerning the invertibility of the process model we refer the reader to [102, 167]. Even if a unique optimum exists, the inverse problem (45)–(49) to determine from given measurements $\boldsymbol{z}$ the quantities $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{v}, \boldsymbol{w}$ might not be well–posed in the sense of Hadamard [77, 117] with respect to continuity in L_2. In fact, a small perturbation of $\boldsymbol{z}$ with respect to the L_2–norm with high frequency oscillations would force $\boldsymbol{y}$ to oscillate as well. In view of (46), (47) this may cause $\boldsymbol{x}$ to have large derivatives which gives rise to arbitrarily large variation of $\boldsymbol{w}$ in L_2 [25]. Consequently, the solution operator $\boldsymbol{T} : (L_2)^{n_y} \longrightarrow (H^1)^{n_x} \times (L_2)^{n_y+n_v+n_w}$ induced by (45)–(49) may not be continuous.

Regularization can be used to guarantee continuity as well as uniqueness. For example, regularization of Tikhonov type includes $\boldsymbol{v}$ and $\boldsymbol{w}$ as quadratic terms in the cost functional [174]. Then, the extended problem formulation

$$\min_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{v},\boldsymbol{w}} \int_{t_{k-m}}^{t_k} \left\{ (\boldsymbol{y}-\boldsymbol{z})^T \boldsymbol{Q} (\boldsymbol{y}-\boldsymbol{z}) + \boldsymbol{v}^T \boldsymbol{R}_{\boldsymbol{v}} \boldsymbol{v} + \boldsymbol{w}^T \boldsymbol{R}_{\boldsymbol{w}} \boldsymbol{w} \right\} \, d\tau \tag{50}$$

with constraints (46)–(49) is well–posed. Here, the regularization parameters $\boldsymbol{R}_{\boldsymbol{v}}, \boldsymbol{R}_{\boldsymbol{w}}$ are time–invariant penalty matrices. Nevertheless, more general weights like e.g. time dependent operators are possible, too. Here, the weights are interpreted in a deterministic sense and do not depend on statistical assumptions. Obviously, an increase of the weights improves stability of the estimates but causes also a growing approximation error in case of non–vanishing $\boldsymbol{v}$, $\boldsymbol{w}$ due to bias. Regularization can also be achieved by projection of (45)–(49) into finite dimensional subspaces, e.g. using a Petrov–Galerkin scheme [151]. No regularization parameter need to be present explicitly. However, there is hidden regularization introduced by the associated discretization. The data error increases when refining the discretization (weak regularization) while a too coarse problem discretization (strong regularization) leads to large regularization errors [77, 117]. Of course, regularization by projection might be combined with direct regularization methods based on penalty terms such as in (50).

Immediately the question arises how to choose the regularization parameters in order to achieve a good compromise between data and regularization error. Optimal parameter selection strategies are available if the noise level of the measurements and the smoothness of the exact solution is known [146]. However, since this information is not available for most practical problems we examined in [25] a strategy, namely the L–curve criterion [93], that does not require this type of knowledge. The approximately best compromise is determined by relating the residual norms under a systematic variation of parameters to a (semi)–norm of the approximation itself. The procedure seems to be well suited for a low number of corrections $\boldsymbol{w}$. However, the problem of quantitatively assessing the best compromise becomes increasingly difficult for a growing number of unknown functions $\boldsymbol{w}$.

In [25] we explore the performance of regularization by projection of (45)–(49) for a simple linear model. The presented refinement concept developed in the context of real–time estimation is here successfully applied to automatically construct sequences of discretization meshes based on upgrades of the approximation spaces. Using the L–curve criterion the refinement process is stopped when a good compromise between data and regularization errors in the estimates is accomplished. Uniform as well as non–uniform problem adapted upgrades of the approximation spaces are employed using techniques outlined in more detail in Section 4. In that particular example it turned out, that non–uniform approximations lead to estimates whose quality cannot be achieved by employing only a uniform mesh. The analysis of ill–posedness presented in [25] can be also extended to problems with nonlinear models.

3 Wavelets

Apart from boundary adaptation wavelet functions $\psi_{j,k}$ are obtained by dilation and translation of a suitable *mother wavelet* ψ, i.e. $\psi_{j,k} = 2^{j/2}\psi(2^j \cdot -k)$, where the normalizing factor $2^{j/2}$ keeps the L_2–norm of the $\psi_{j,k}$ independent of j, k. Thus when ψ is chosen to have bounded support, as it will always be the case in our applications, one has $\mathit{diam\,supp}\,\psi_{j,k} \sim 2^{-j},\ j \to \infty$. The key feature is that for suitable ψ the collection $\boldsymbol{\Psi} = \{\psi_{j,k}\}$ forms a *Riesz basis* of L_2, i.e., every $f \in L_2$ has a unique expansion

$$f = \boldsymbol{\Psi}^T \boldsymbol{d}_f = \sum_{j,k} d_{j,k}\psi_{j,k}$$

and the norm equivalence

$$c_1 \parallel \boldsymbol{d}_f \parallel_{\ell_2} \leq \parallel f \parallel_{L_2} \leq c_2 \parallel \boldsymbol{d}_f \parallel_{\ell_2} \tag{51}$$

holds for some positive constants c_1, c_2 independent of f. In the special case when the $\psi_{j,k}$ form an orthonormal basis, equality holds between coefficient and function norm. In general, (51) implies the existence of a *dual basis* $\tilde{\boldsymbol{\Psi}}$ in L_2 which is *biorthogonal* to $\boldsymbol{\Psi}$, i.e.

$$\langle\langle \boldsymbol{\Psi}, \tilde{\boldsymbol{\Psi}} \rangle\rangle = \boldsymbol{I} \tag{52}$$

where we use the shorthand notation $\langle\langle \boldsymbol{\Psi}, \tilde{\boldsymbol{\Psi}} \rangle\rangle = (\langle \psi_{j,k}, \psi_{i,l} \rangle)_{(j,k),(i,l)}$ and $\langle \cdot, \cdot \rangle$ is the standard L_2 inner product. Thus the wavelet coefficients $d_{j,k}$ are given by $\langle f, \tilde{\psi}_{j,k} \rangle$.

Clearly, (51) means that small perturbations of the expansion coefficients $\boldsymbol{d}_f$ cause only small changes of f in the L_2–norm and vice versa. In particular, retaining only the first N largest coefficients in modulus provides –up to a uniform constant– the best approximation to f that can be composed from any selection of N basis functions. This is the basis for nonlinear compression techniques (*best N–term approximation*). Accordingly the objective of *adaptive* approximation is to successively track the most significant coefficients of the function to be determined.

Furthermore, depending on the smoothness of the employed wavelets also norm equivalences with respect to other function spaces such as Sobolev and Besov spaces hold with appropriate scale dependent weights on the wavelet coefficients. They are essential for signal analysis (see Section 5) and for the preconditioning of discretized differential and integral operators (see Subsection 6.4).

The starting point for the construction of wavelet bases is usually a so–called *multiresolution sequence* of nested spaces $S_{j_0} \subset \ldots \subset S_j \subset S_{j+1} \subset \ldots$ whose union is dense in L_2. Here j_0 stands for some coarsest discretization level. The spaces S_j are spanned by scaling functions, $S_j = \operatorname{span}\{\varphi_{j,k} : k \in \mathcal{I}_j\}$, which are obtained by a suitable, compactly supported function φ. Due to the compact support of φ we can view 2^{-j} as the uniform mesh size of S_j. Given a suitable dual sequence $\tilde{S}_j$, a successive decomposition of S_j leads to the multiscale splitting $S_J = S_{j_0} \bigoplus_{j=j_0}^{J-1} W_j$ where $W_j = \operatorname{span}\{\psi_{j,k} : k \in \mathcal{I}'_j\} \perp \tilde{S}_j$. Analogous splittings for the spaces $\tilde{S}_j$ yield complement spaces $\tilde{W}_j \perp S_j$. The bases $\boldsymbol{\Psi}$ and $\tilde{\boldsymbol{\Psi}}$ for the spaces W_j respectively $\tilde{W}_j$ satisfy (52).

All properties above can be realized when choosing φ as the cardinal B–spline of order m. Moreover, one can construct for any $\tilde{m} \in \mathbb{N}, \tilde{m} \geq m, m + \tilde{m}$ even, a compactly supported dual scaling function $\tilde{\varphi}$ such that the dual multiresolution spaces $\tilde{S}_j$ contain all polynomials up to order $\tilde{m}$. As a consequence of biorthogonality, one has the *moment conditions*

$$\langle (\cdot)^r, \psi \rangle = 0, \qquad r = 0, \ldots, \tilde{m} - 1 \,. \tag{53}$$

Likewise one has $\langle (\cdot)^r, \tilde{\psi} \rangle = 0$ for $r = 0, \ldots, m - 1$. Therefore, smooth functions have wavelet expansions with rapidly decaying wavelet coefficients so that levelwise truncation of wavelet expansions provide good approximations. However, as soon as sharp transitions or even singularities occur different nonuniform selections of basis functions are expected to provide more economical approximations. In fact, combining the above statements with the locality of wavelets, large wavelet coefficients reflect a large local change of the function.

4 Discretization: Wavelets as Trial and Test Functions

For the discretization of the optimization problem on a finite time horizon $[t_{k-m}, t_k]$ we scale the horizon to $[0,1]$ and formulate the equality constraints (46), (47) in a weak sense. We obtain the equality constraints:

$$\langle \dot{\boldsymbol{x}} - \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}) - \boldsymbol{W}\boldsymbol{w}, \boldsymbol{\nu}_1 \rangle \;=\; \boldsymbol{0} \quad \text{for all } \boldsymbol{\nu}_1 \in L_2^{n_x} \tag{54}$$

$$\langle \boldsymbol{y} - \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}) - \boldsymbol{V}\boldsymbol{v}, \boldsymbol{\nu}_2 \rangle \;=\; \boldsymbol{0} \quad \text{for all } \boldsymbol{\nu}_2 \in L_2^{n_y} \,. \tag{55}$$

Discretization is then given by the representation of each function with respect to an appropriately chosen wavelet basis $\boldsymbol{\Psi}$, i.e. we represent $\boldsymbol{x}$ as $\boldsymbol{x} = \boldsymbol{\Psi}_{\Lambda_x}^T \boldsymbol{d}_{\boldsymbol{x}}$, $\boldsymbol{y}$

as $\boldsymbol{y} = \boldsymbol{\Psi}^T_{\Lambda_y} \boldsymbol{d}_{\boldsymbol{y}}$ etc.. We obtain an equivalent, *infinite dimensional* but *discretized* optimization problem for the wavelet coefficients $\boldsymbol{\zeta}_\Lambda = (\boldsymbol{d}^T_{\boldsymbol{x}}, \boldsymbol{d}^T_{\boldsymbol{y}}, \boldsymbol{d}^T_{\boldsymbol{v}}, \boldsymbol{d}^T_{\boldsymbol{w}})^T$:

$$\min_{\boldsymbol{\zeta}_\Lambda} \; \boldsymbol{\zeta}^T_\Lambda \boldsymbol{H}_\Lambda \boldsymbol{\zeta}_\Lambda \quad + \quad \boldsymbol{b}^T_\Lambda \boldsymbol{\zeta}_\Lambda \tag{56}$$

$$\text{s.t.} \qquad \boldsymbol{F}_\Lambda(\boldsymbol{\zeta}_\Lambda) \;=\; \mathbf{0} \tag{57}$$

$$\boldsymbol{n}_\Lambda(\boldsymbol{\zeta}_\Lambda) \;\leq\; \mathbf{0}\,. \tag{58}$$

In our particular case we use piecewise linear, continuous wavelets, (i.e the scaling function is a B–spline of order $m = 2$ fulfilling the moment conditions with $\tilde{m} = 2$), as trial functions for $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{v}$ and for the test functions $\boldsymbol{\nu}_2$, while we employ piecewise constant wavelets ($m = \tilde{m} = 1$) for $\boldsymbol{w}$ and the test functions $\boldsymbol{\nu}_1$. This corresponds to the minimal regularity for a conforming discretization.

While the above formulation yields still the exact solution, for the numerical treatment we have to choose finite dimensional approximations. Hence, for each function involved we have to choose a finite set of basis functions, described by their index set $\Lambda_{\boldsymbol{x}}$ etc.. As mentioned before, we will start with some initial possibly small index sets $\Lambda^0_{\boldsymbol{x}}$ etc. and *adaptively* enlarge these index sets to $\Lambda^{\ell+1}_{\boldsymbol{x}} = \Lambda^{\ell}_{\boldsymbol{x}} \cup \Gamma_{\boldsymbol{x}}$ in each refinement step. Due to the piecewise linear basis functions the inequality constraints are guaranteed to be satisfied exactly, by enforcing their validity at the corresponding mesh points.

5 Signal Processing

The tasks to be performed in the signal processing part are transformation of the discrete measurement samples into a continuous representation, possibly continuation of missing signal parts, denoising and outlier removal, as well as data compression. Recent developments show that wavelets are very well suited for dealing with these tasks [41, 69, 72, 73]. Hence, we restrict ourselves to mentioning only the basic properties of wavelets employed for signal analysis, namely, the norm equivalences with respect to Besov spaces and the vanishing polynomial moments mentioned in Section 3. They give rise to nonlinear approximation techniques based on efficient threshold and/or shrinkage algorithms. However, the choice of the involved parameters depends typically on statistical model assumptions, such as white Gaussian noise with a given noise level. To our knowledge, theoretical results under less stringent assumptions do not exist. The current state of our algorithm is based on data fitting on dyadic meshes $\{2^{-J}k\}$ and a shrinkage algorithm based on the results of [41]. The data are considered as scaling function coefficients of a correspondingly high level of resolution. To this data format one can then apply the wavelet transform. In case of nonuniform sampling rates a more sophisticated fitting procedure is needed. Considering a single function z_i and omitting the index i, we obtain the wavelet representation $z^\delta = \boldsymbol{\Psi}^T \boldsymbol{d}^\delta_z$. Corresponding to the trial functions for the outputs $\boldsymbol{y}$, we choose piecewise linear, continuous wavelets for this purpose. Then, to obtain the denoised signal functions $z = \mathbf{\Psi}^T \boldsymbol{d}_z$ we apply the shrinkage algorithm (omitting the index z) $d_{j,k} := d^\delta_{j,k} - sign(d^\delta_{j,k})\,\varepsilon$ for $|d^\delta_{j,k}| \geq \varepsilon$ and $d_{j,k} := 0$ otherwise, where ε depends only on the noise level. If little or nothing is known about the noise, ε has to be tuned in a rather heuristic manner, usually depending on the magnitude of the wavelet coefficient, the location k and the scale j. For a more detailed description we refer the reader to [20].

The representation of the measurements in terms of wavelets is fully compatible with our multiscale discretization of the constrained optimization problem. In particular, since the best N–term wavelet approximation is essentially determined by the first N largest (in modulus) wavelet coefficients, the initial index set Λ_0 for discretizing the optimization problem is easily identified. Only these coefficients will be used in the first step. In further refinement steps of the optimization problem additional wavelet coefficients of $\boldsymbol{z}$ will be processed. For the refinement of the employed basis index set, which is based on the measurements as well as on the current approximation of the estimates, we refer to Section 6.1.

6 The Optimization Problem on a Fixed Horizon

To realize the overall concept of multiscale moving horizon optimization under the particular real time requirement, we encounter many new problems on each level of the solution process even for an optimization problem on a fixed horizon. The sketch in Figure 9 indicates the main conceptual blocks of our approach on a fixed horizon, respectively its current state of development.

In the subsequent subsections we shall briefly describe the main ingredients listed in Figure 9. The outermost level corresponds to the discretization of the whole optimization problem and the adaptive refinement. The goal is to spend minimal computational effort for realizing a fixed decay rate of the current error in each refinement step. The next subsection will briefly present our algorithm and some numerical results. Then, the treatment of nonlinearities will be presented. The optimization problem arising on each level of the resolution hierarchy is currently solved by a general purpose algorithm based on a SQP–method. Some algorithmic details are given in 6.2. Ultimately, the general purpose SQP–solver is to be replaced by a fully problem adapted scheme. A core ingredient is the treatment of the corresponding linearized problems. Here, the outer loop is concerned with the treatment of the inequality constraints. We address the question whether interior–point methods or active–set methods can exploit the information of the previous refinement step in a more efficient way. Results are given in Subsection 6.3. Finally, the solution process for the arising linear systems is outlined in 6.4. The computational cost for a fixed error reduction should remain proportional to the current size of the problem. This immediately suggests the idea to use an iterative solver rather than a direct solver. The current state of development is briefly described. Also preconditioning and stopping criteria are discussed in the context of nested iteration.

Figure 9: Outline of the conceptual blocks on a fixed horizon.

Dynamic optimization problem
adaptive refinement employing
– sensitivity of the cost functional with respect to $\boldsymbol{w}$ and $\boldsymbol{v}$
– local error estimator for the differential equations

NLP
treatment of the nonlinearities
– general purpose algorithm based on a SQP–method
– efficient evaluation of $\langle f(\boldsymbol{\Psi}_\Lambda^T \boldsymbol{d_x}), \psi_{j,k} \rangle$ for given $\boldsymbol{d_x}$

QP
treatment of the inequality constraints
– interior–point methods for the first approximations
– active–set methods with initialization of the active sets for the refinement steps

Linear system of equations
iterative solver (simplified Uzawa)
– employing previous solution as initial value
– using adapted Jacobi preconditioner motivated by wavelet theory

6.1 Adaptive Refinement

An adaptive discretization strategy is essential for an efficient treatment of the optimal control problem. In our refinement approach we start with a coarse approximation with relatively few trial functions collected in Λ^0. Then, based on the information of the previously computed approximation $\boldsymbol{d}_\Lambda^{*\ell}$ at refinement step ℓ an improved index set $\Lambda^{\ell+1}$ is generated and (56)–(58) is resolved. Algorithmic details on the adaptive refinement strategy are given in [23, 21]. Therefore, we present here only the underlying ideas.

The **refinement of the known quantities $\boldsymbol{z}$ and $\boldsymbol{u}$** is only discussed for a single function z_i, omitting the index i for convenience, since everything applies to any component in the same way. We are interested in compressed approximations of the measurements z_Λ^ℓ satisfying

$$\| z - z_\Lambda^\ell \|_{L_2} \;\leq\; \varepsilon'_\ell \| z \|_{L_2} \tag{59}$$

with given tolerance ε'_ℓ. As mentioned in Section 3 the N largest wavelet coefficients of z give up to a fixed constant factor the best N–term approximation. Hence, given the wavelet coefficients $\boldsymbol{d}_z$ of the full but finite expansion $z = \boldsymbol{\Psi}_\Lambda^T \boldsymbol{d}_z$, we simply have to neglect the elements of $\boldsymbol{d}_z$ whose moduli are below a certain threshold ε which uniquely depends on ε'_ℓ. The remaining entries are the *significant* coefficients and their indices form the index set Λ_z^ℓ. Typically the number of significant wavelet coefficients is by far smaller than the number of discrete measurement samples. During the refinement sequence the approximation quality is increased according to $\varepsilon'_{\ell+1} < \varepsilon'_\ell$.

The **refinement of states $\boldsymbol{x}$ and outputs $\boldsymbol{y}$** is based on an error analysis of $\boldsymbol{x}_\Lambda^\ell$ only, since errors in $\boldsymbol{y}_\Lambda$ are directly linked to errors in $\boldsymbol{x}_\Lambda$. Residual based error analysis is computationally inexpensive and gives usually a qualitatively good grasp on the error behavior. However, sharp error estimates which are needed to efficiently treat index problems arising from active state inequality constraints are difficult to obtain [109]. Therefore, the refinement is based on local error estimation where $\boldsymbol{x}_\Lambda^{*\ell}$ is compared to *locally* refined solutions keeping $\boldsymbol{w}_\Lambda^{*\ell}$ at their current optimal values. In particular, we use a local trapezoidal rule and evaluate the error at the midpoints of the current mesh determined by Λ_x^ℓ. Where the error bounds are violated, we increase the set of wavelet basis functions by adding to the current set those basis functions with neighboring indices in the time frequency plane.

In contrast to the local error estimator used for the states $\boldsymbol{x}$ and output functions $\boldsymbol{y}$, the problem formulation allows us to employ the sensitivity of the Lagrange functional for the **refinement of unknown inputs $\boldsymbol{v}$ and $\boldsymbol{w}$**. Hence, for an improved approximation of w_i we evaluate the gradients of the Lagrange functional with respect to *potentially new* trial functions for w_i determined by the index set $\Pi_{w_i}^\ell$. Local neighbors of the indices in $\Lambda_{w_i}^\ell$ are excellent *candidates* for $\Pi_{w_i}^\ell$ where we require $\Pi_{w_i}^\ell \cap \Lambda_{w_i}^\ell = \emptyset$. Once the gradients are computed, the trial functions associated with the larger absolute gradients are interpreted as those basis functions which have large impact on the solution and therefore should be used for refinement. The smaller ones are neglected. The same approach is applied to $\boldsymbol{v}$.

The approximations for $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{v}, \boldsymbol{w}$ might change during the cycles of refinement such that previously needed trial functions may become obsolete. The **elimination of unnecessary trial functions** is based on a compression technique as outlined before. Basis functions corresponding to significant wavelet coefficients are kept while the small ones are discarded.

Finally the different index sets for the various functions have to be combined to take their close interactions into account.

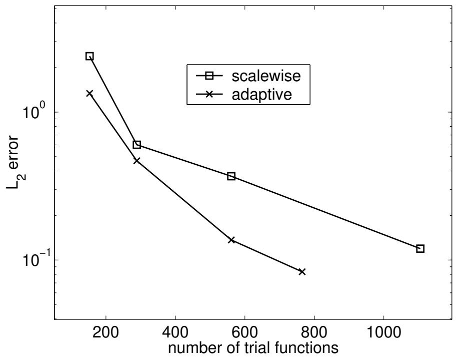
In summary, the overall **adaptive refinement algorithm** proceeds as follows denoting by Γ^ℓ the set of newly added indices:

1. Solve problem (56)–(58) with the index set Λ^ℓ.

2. Refine $\boldsymbol{z}, \boldsymbol{u}$ by *thresholding* $\rightarrow \Gamma_z^\ell, \Gamma_u^\ell$.

3. Apply a *local error* estimator for $\boldsymbol{x}$, $\boldsymbol{y}$ $\longrightarrow \Gamma_{x_i}^\ell, \Gamma_{y_i}^\ell$.

Figure 10: Scalewise versus adaptive refinement.



4. Use the *sensitivity* of Lagrange functional for $\boldsymbol{v}$, $\boldsymbol{w} \longrightarrow \Gamma^{\ell}_{v_i}, \Gamma^{\ell}_{w_i}$.

5. *Compress* current approximation by thresholding $\longrightarrow \tilde{\Lambda}^{\ell}_{x_i}$ etc.

6. Take *interactions* into account $\longrightarrow \Gamma^{\ell}$

7. Form $\Lambda^{\ell+1} = \tilde{\Lambda}^{\ell} \cup \Gamma^{\ell}$.

8. Set $\ell := \ell + 1$, go to 1.

Figure 10 illustrates for a typical example the comparison between different choices of the index sets Λ^{ℓ}, i.e. a uniform refinement which adds all basis functions of the next scale and an adaptive refinement which is based on the considerations above. The L_2 error is plotted versus the number of trial functions. We see, for example, that for an error 0.105 we need roughly 550 basis functions with an adaptive approach while approximately 1000 degrees of freedom arise for a uniform mesh (mesh size 2^{-10}). Obviously adaptivity outperforms uniform discretization with regard to the cost for computing an approximation for a given target accuracy.

6.2 Solving the NLP

In principle, problem (56)–(58) can be solved with any available NLP method like sequential quadratic programming (SQP) or generalized Gauss–Newton methods. The latter ones are particularly attractive when the cost functional is close to zero [161]. However, to achieve highest possible efficiency the structure of the problem formulation, i.e. the sparsity pattern of the model equations system, as well as the structure of the discretization should be exploited to their full extent. Nevertheless, for nonlinear problems we currently employ only a general purpose SQP–method designed for solving large sparse NLP problems based on an active set strategy. For the specific case of linear dynamic models a tailored strategy exploiting the discretization structure is presented in more detail in Subsection 6.4 .

A general purpose SQP–method such as SNOPT [87] typically requires routines to evaluate the cost functional in (56) and the residuals of the restrictions (57),

(58) at a current iterate $\boldsymbol{\zeta}^{\ell,i}$ as well as their first order derivatives with respect to $\boldsymbol{\zeta}^{\ell,i}$. In order to provide these data, inner products of wavelet functions have to be computed. Inner products of linear terms in the equations (57), (58) do not depend on current iterates and therefore can be computed prior to the optimization run with known numerical techniques for a sufficiently large index set Λ. Then, in each refinement step ℓ those rows and columns of the precomputed and stored inner product matrix whose indices are contained in Λ^ℓ have to be retrieved only from these files.

The computation of nonlinear terms in the optimization problem is more difficult and costly since we have to evaluate inner products of the form $\langle f(\boldsymbol{x}_\Lambda^{\ell,i}), \psi_{j,k}\rangle$ where $\boldsymbol{x}_\Lambda^{\ell,i}$ is given in a wavelet expansion and f is a nonlinear function. Straightforward quadrature would spoil the complexity gain by adaptive refinements such that more sophisticated efficient evaluation schemes are needed. Recently, Dahmen *et al.* [62] developed a methodology to efficiently evaluate such inner products which avoids the complexity of the finest uniform mesh and requires only a computational effort proportional to the dimension of the adapted wavelet basis.

A simple but efficient warmstart of the SQP–method is obtained by reusing the solution in refinement step $\ell - 1$ as initial guess for the solver in step ℓ. Due to the hierarchical structure of the wavelet basis a change of basis is not necessary. The initial values for the unknowns $\boldsymbol{\zeta}^{\ell,0}$ as well as for the Lagrange multipliers are efficiently provided by $d_{j,k}^{\ell,0} = d_{j,k}^{*\ell-1}$ for $(j,k) \in \Lambda^{\ell-1} \cap \Lambda^\ell$ and $d_{j,k}^{\ell,0} = 0$ otherwise. Additional warm start strategies like updating the Hessian matrices or their approximations may also reduce the numerical cost but have not been investigated yet for the nonlinear problem.

6.3 Solving the QP

For the treatment of the quadratic programming problems, which arise either as subproblems of the nonlinear programming problem or directly for dynamic optimization problems with linear process model and quadratic cost functional, several decisions concerning the solution strategy have to be made. Typically, a large number of algebraic inequalities is present due to the discretization of the inequality restrictions (58). These inequality restrictions have to be handled efficiently and a method is required which takes full advantage of the refinement concept.

Two common approaches based on active sets (AS) and on an interior point (IP) method have been compared. On one hand, IP methods are often applied to systems with a very large number of inequalities. They usually need a fixed number of iterations independently of the dimension of the system [190]. Hence they suggest themselves in the present context. On the other hand, AS methods need to solve smaller linear systems in each iteration step. Nevertheless, the number of iterations typically increases with the number of inequality constraints. This typically happens if one solves the optimization problem in a single run. In our refinement algorithm, though, this will be the case only in the first loop. For the next refinement we can employ the information from the previous step. In case of the IP method we can use the solution as initial guess for the refined optimization problem. However, the central idea of the IP is to drive all quantities simultaneously to the bounds at approximately the same rate [190]. Hence, even if a good initial guess is taken, some of the variables might be too close to the boundary and a centering step becomes necessary. The next iterate presumably moves away from the optimum. Therefore, the warmstart potential of IP methods seems to be limited in our context. In contrast, in case of AS methods we can initialize the active set on refinement level $\ell + 1$ with the set for which the current solution $\boldsymbol{\zeta}^{\ell}$ is active on the refined mesh. Since the current solution should be a good approximation of the refined solution, we do not expect significant changes of the active set and, therefore, we do not expect a large increase of iteration numbers.

These considerations are confirmed by numerical experiments reported in [22, 186]. Therefore, our strategy is to use an **interior–point** method for the **first** approximate solution of the optimization problem and for all **following** refinements we use the **active–set** method **with initialization** based on the previous solution.

6.4 Solving the Linear System

Since even for nonlinear problems the same issues would arise after linearization we confine the discussion here for the sake of clarity to a linear model problem. Moreover, the application of the active–set method leads to saddle point problems of the form:

$$\boldsymbol{\mathcal{L}}_\Lambda \begin{pmatrix} \boldsymbol{\Upsilon}_\Lambda \\ \boldsymbol{\lambda}_\Lambda \end{pmatrix} := \begin{pmatrix} \boldsymbol{\mathcal{A}}_\Lambda & {\boldsymbol{\mathcal{B}}_\Lambda}^T \\ \boldsymbol{\mathcal{B}}_\Lambda & \mathbf{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{\Upsilon}_\Lambda \\ \boldsymbol{\lambda}_\Lambda \end{pmatrix} = \begin{pmatrix} \boldsymbol{h}_\Lambda \\ \mathrm{g}_\Lambda \end{pmatrix}, \tag{60}$$

which are structurally the same as if only equality constraints are present in the optimization problem. Hence, in order to focus on the treatment of such linear systems we consider in a first step the following optimization problem

$$\min_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{v},\boldsymbol{w}} \int_0^1 \left\{ (\boldsymbol{y}-\boldsymbol{z})^T \boldsymbol{Q} (\boldsymbol{y}-\boldsymbol{z}) \quad + \quad \boldsymbol{v}^T \boldsymbol{R}_v \boldsymbol{v} + \boldsymbol{w}^T \boldsymbol{R}_w \boldsymbol{w} \right\} \, d\tau \tag{61}$$

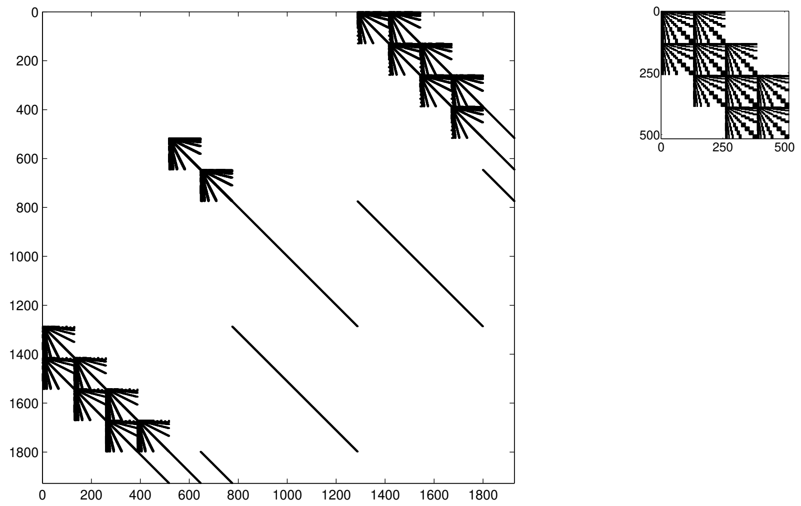
$$\text{subject to} \qquad \dot{\boldsymbol{x}} - \boldsymbol{A}\boldsymbol{x} - \boldsymbol{W}\boldsymbol{w} \;=\; \boldsymbol{B}\boldsymbol{u} \tag{62}$$

$$\boldsymbol{y} - \boldsymbol{C}\boldsymbol{x} - \boldsymbol{V}\boldsymbol{v} \;=\; \mathbf{0}\,. \tag{63}$$

However, the ideas can be extended to the nonlinear case with inequality constraints. Figure 11a shows the typical structure of the Karush–Kuhn–Tucker matrix $\boldsymbol{\mathcal{L}}_\Lambda$ for an example of fourth order. It exhibits the block structure, where each block corresponds to entries of the regularization and model matrices. The blocks are either diagonal or have finger structure arising from wavelet scalar products.

The discretization error is bounded from below by the approximation error of the functions in the adaptively chosen wavelet space. Hence, it is sufficient to solve

Figure 11: a) KKT–matrix $\boldsymbol{\mathcal{L}}_\Lambda$ b) Schur complement $\boldsymbol{\mathcal{S}}_\Lambda$.



the above system of equations only up to an error of this order. The goal is now to use an efficient solver up to an *accuracy* which *corresponds* to the *discretization error on each refinement level*. For efficiency reasons, it is important to exploit the previous approximation. Furthermore, successive error reduction is in the spirit of the overall concept. Hence, while most optimization solvers use direct methods in this context we focus on **iterative solvers**. Of course, then the discretization matrices do not have to be assembled but only their application to a vector has to be guaranteed. Due to the indefiniteness of the system one could think of using *gmres*, or, exploiting the symmetry, of using *symmlq* or *minres* [9, 134]. Nevertheless, to exploit the problem structure as much as possible, we **reduce the system** and obtain a matrix $\tilde{\boldsymbol{\mathcal{L}}}_\Lambda$, where we can apply the Uzawa algorithm (see [35]). This algorithm is particularly designed for saddle point problems with positive definite $\boldsymbol{\mathcal{A}}_\Lambda$. Conceptionally it applies the *pcg*–method to the Schur complement and, to avoid inversion of $\boldsymbol{\mathcal{A}}_\Lambda$, also to $\boldsymbol{\mathcal{A}}_\Lambda$. In our case the inner application of the *pcg*–method is not necessary, since $\boldsymbol{\mathcal{A}}_\Lambda^{-1}$ can actually be set up very efficiently. This means, that we apply essentially the *pcg*–method to the Schur complement $\boldsymbol{\mathcal{S}}_\Lambda$ of the reduced system. For a detailed discussion of this **adapted Uzawa algorithm** we refer to [24]. In Figure 11b we can see the structure of the linear system after a reduction of $\boldsymbol{\mathcal{L}}_\Lambda$ to the Schur complement $\boldsymbol{\mathcal{S}}_\Lambda$. The problem size is reduced from a size corresponding to all functions $\boldsymbol{x}$, $\boldsymbol{y}$, $\boldsymbol{v}$, $\boldsymbol{w}$ and the Lagrange parameter functions $\boldsymbol{\lambda}_1$, $\boldsymbol{\lambda}_2$ to the size of the discretization of $\boldsymbol{x}$. Furthermore, $\boldsymbol{\mathcal{S}}_\Lambda$ is still very sparse and has finger structure. Nevertheless, it will never be set up explicitly to avoid matrix multiplications.

As for the application of iterative methods there arise, of course, two essential questions: preconditioning and stopping criteria. These two questions will be discussed next in the context of nested iteration, which is applied to the sequence of equations

$$\tilde{\boldsymbol{\mathcal{L}}}_{\Lambda^\ell}\begin{pmatrix}\tilde{\boldsymbol{\lambda}}_{\Lambda^\ell}\\ \tilde{\boldsymbol{\Upsilon}}_{\Lambda^\ell}\end{pmatrix}=\begin{pmatrix}\tilde{\boldsymbol{h}}_{\Lambda^\ell}\\ \tilde{\mathbf{g}}_{\Lambda^\ell}\end{pmatrix},\quad \ell=\ell_0,\ell_0+1,\ldots, \tag{64}$$

with increasing dimension corresponding to the refined optimization problems.

The corresponding operator on the infinite dimensional function spaces

$$\tilde{\boldsymbol{\mathcal{L}}}:(L_2)^{n_y+n_w}\times(H^1)^{n_x}\longrightarrow(L_2)^{n_y+n_w}\times((H^1)^{n_x})' \tag{65}$$

is bounded and has a bounded inverse with respect to these appropriate norm. In order to obtain well conditioned discretizations in the Euclidean metric l_2 one exploits the fact that suitably weighted l_2–norms of the wavelet coefficients are equivalent to Sobolev norms, i.e. $\{2^{-sj}\psi_{j,k}\}$ form a Riesz basis of H^s, here for $s\in\{0,1\}$. This corresponds to a symmetric diagonal scaling of the wavelet representation $\tilde{\boldsymbol{\mathcal{L}}}_\Lambda$ of $\tilde{\boldsymbol{\mathcal{L}}}$ with infinite index set Λ. This scaling can also be viewed as preconditioning of $\tilde{\boldsymbol{\mathcal{L}}}_\Lambda$. For our particular case, this leads to a symmetric preconditioning of the Schur complement $\boldsymbol{\mathcal{S}}_\Lambda$ with a diagonal matrix $\boldsymbol{M}^{1/2}_{\Lambda^\ell}$ with scale dependent entries 2^{-j}. The boundedness of the preconditioned $\tilde{\boldsymbol{\mathcal{L}}}_\Lambda$ and its inverse remain valid for finite index sets Λ^ℓ independently of ℓ provided that the Galerkin scheme associated with these trial spaces is stable, i.e. the trial spaces satisfy the LBB–condition [35]. Consequently, we can deduce that the condition numbers fulfill

$$\kappa_\ell:=\mathrm{cond}_2(\boldsymbol{M}^{1/2}_{\Lambda^\ell}\boldsymbol{\mathcal{S}}_{\Lambda^\ell}\boldsymbol{M}^{1/2}_{\Lambda^\ell})\leq c \tag{66}$$

independently of ℓ. The convergence rate of the adapted Uzawa algorithm can be estimated in terms of κ_ℓ and, therefore, is also bounded independently of the refinement step.

Instead of the diagonal entries 2^{-2j} in $\boldsymbol{M}$ we use an **approximate Jacobi preconditioner**, i.e. an approximation of the diagonal of the Schur complement $\boldsymbol{\mathcal{S}}_{\Lambda^\ell}$ which can be set up very efficiently. The entries reflect the necessary scale dependent scaling. In addition this preconditioner corresponds to normalizing the wavelet basis in the energy norm and usually gives better results. Numerical experiments confirm the superiority to only scaling by powers of two [24].

The **stopping criteria** for the iterative solver hinges on three properties: (i) the maximal approximation order of the discretization spaces in the energy norm giving a reduction rate α, (ii) the norm equivalences relating the energy norm to the wavelet coefficients , and (iii) the convergence order of the underlying *pcg*–method, which can be estimated in terms of the condition numbers κ_ℓ. These properties can be used to determine an upper bound for the number of iterations

$$max_{it}\approx log\,(2\bar{c}/\alpha)\,/\left(-log\left((\sqrt{\kappa}-1)/(\sqrt{\kappa}+1)\right)\right) \tag{67}$$

and a bound for the maximal relative residual error

$$\alpha_{rel}\approx\alpha/(\bar{c}\sqrt{\kappa}) \tag{68}$$

needed to produce on each level approximate solutions with discretization error accuracy (see [24]). The constant $\bar{c}$ is given by the norm equivalences. For our particular discretization we expect at best a first order convergence in the energy norm. Hence, in terms of the number of degrees of freedom we take $\alpha = \#\Lambda^{\ell-1}/\#\Lambda^{\ell}$, which is 1/2 for uniform mesh refinement. Moreover, to determine α_{rel} and max_{it} an upper bound for κ has to be estimated. As a pragmatic choice we used $\alpha_{rel} = 10^{-4}$ and $max_{it} = 20$ in our experiments, which was sufficient in most cases.

In spite of the accomplished asymptotic boundedness, large condition numbers may still arise due to system inherent features as for example, a poor observability measure, stiff differential equations or inadequate regularization. The effect of the choice of the regularization parameter is studied in [24] by means of numerical experiments. A rather weak regularization of the model error functions $\boldsymbol{w}$ motivated by stabilizing the original ill–posed problem of state estimation turns out to be favorable for the condition numbers of the systems considered here. Nevertheless, conclusive statements about the quantitative effect of regularization and of system inherent features on the iterative schemes can not be made, yet.

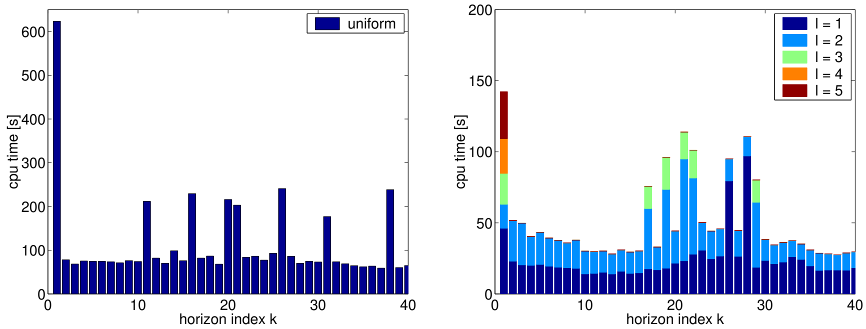
While the condition numbers provide an upper bound for the iteration numbers, the actual eigenvalue distributions give rise to considerably smaller iteration numbers. Superconvergence, known for the cg–method, can be observed for the adapted Uzawa algorithm, too. Nevertheless, more sophisticated methods have to be developed to affect the smallest eigenvalues in a more drastic way. Currently, a Schur complement method based on the adaptive refinement algorithm is under investigation.

First numerical results for the algorithm above can be found in [24]. It turns out that depending on the system inherent features, the nested iteration process is more efficient than solving the fully discretized system directly. Perhaps more important than the expected higher efficiency of nested iteration is the fact it provides successively improved approximations at a much earlier stage. We can see also that it is not necessary to determine on each scale the exact solution. The discretization error on each refinement level is nearly reached, although the residuals are still quite large, in full agreement with the size of the condition numbers.

7 Moving Horizon Optimization

So far we have outlined the refinement approach for the optimization on a fixed horizon. In a moving horizon framework quantitative information about the solution, its structure and the corresponding mesh is available from the last horizon $[t_{k-m-1}, t_{k-1}]$. This information is to be exploited on the next horizon. The approximate solution on the $k-1$–th horizon is extrapolated in step k to the non–overlapping window interval $[t_{k-1}, t_k]$ and employed as initial guess ($\ell = 0$) for the optimization problem. For simplicity we employ linear extrapolation. Then, the approximation on the k–th horizon is expressed in its wavelet expansion. In order to ensure that the refinement

Figure 12: CPU times in each horizon. Uniform and adaptive discretization.



procedure has time to prompt an approximate solution, signal based compression (see Section 5) of all quantities with a suitable compression rate provides an initial index set $\Lambda^{0,k}$ for the optimization problem on the k–th horizon.

We applied the refinement approach presented to a well known literature example (see e.g. [66]). The process consists of an ideal continuous stirred tank reactor where a reversible exothermic reaction $A \;\mathbf{\Omega}\; R$ is carried out. The objective in the example is to move the system from an equilibrium point of low conversion and high temperature to a target equilibrium point of low temperature and significantly higher conversion. This optimal control problem is approached using an MPC framework and employing a quadratic cost functional. The horizon length is chosen to be $T = 400\ min$ and the horizon is shifted every $\Delta T = 10\ min$. Details on the model equations and parameters used can be found in [22]. The optimization problem is solved employing the active set based NLP solver SNOPT [87] on a Sun Ultra 2 (167 MHz) workstation. The numerical effort on each horizon needed to compute solutions of comparable accuracy for one uniform mesh discretization (scale 7) and for the adaptive refinement approach is shown in Figure 12. For the latter we used a mesh relaxation factor of $\eta = 0.7$. Note, that both figures are scaled differently. The numerical work with and without warm start favor the adaptive refinement approach. The gain is particularly large in the very first horizon where no solution is available for initialization. Moreover, the intermediate solutions given by the refinement approach provide backup estimates for the real time restrictions. They might also be directly applied to the real process at an earlier time.

8 Conclusions

We have introduced a multiscale concept for moving horizon optimization based on wavelets in order to meet real–time requirements. Of course, the development and validation of the presented concept is a rather complex task. The information obtained on the current level of discretization has to be exploited on the next hierarchy level. It is used to identify a suitable refinement, to possibly adjust the regularization, and to speed up the solution of the refined discretization of the estimation or control problem. So far we investigated, although on different levels of depth, the following components of the problem: regularization, input processing, optimization on a fixed horizon as well as moving the horizon. The relevant ingredients are now essentially available and the results indicate the potential of this approach.

However, we had to face in the development of these components a diversity of problems, some of which do only occur due to this particular real–time requirement and the multiscale ansatz whereas others are problem inherent to process monitoring. Not all of these obstructions could be overcome in a satisfactory way yet. Further investigations should include for example the combinations of direct and indirect regularization techniques presented, as well as level dependent Tikhonov–type regularization and additional regularization of the states. Also the influence of the length of the horizon on the estimation quality for given noisy data has to be investigated more thoroughly. The signal processing part has to be extended to non uniform sampling and colored noise. For the treatment of the optimization on a fixed horizon, for example, the used general purpose SQP–method should be replaced by a fully problem adapted method. A necessary improvement of the preconditioner for the arising linear systems is currently under investigations. It exploits the multiscale setting and is based on a Schur complement technique.

The developed ingredients still have to be combined to one single software implementation. Moreover, while the numerical examples have been mostly of model character, the optimization software with all its necessary extensions should be finally examined for more realistic industrial problems of large scale under real–time conditions. Of course, the listed future directions are not exhaustive.

On the regularization of dynamic data reconciliation problems[1]

Thomas Binder, Luise Blank, Wolfgang Dahmen, Wolfgang Marquardt

in: J. of Process Control 12, 557–567, 2002.

Abstract

Dynamic data reconciliation problems are discussed from the perspective of the mathematical theory of ill-posed inverse problems. Regularization is of crucial importance to obtain satisfactory estimation quality of the reconciled variables. Usually, some penalty is added to the least-squares objective to achieve a well-posed problem. However, appropriate discretization schemes of the time-continuous problem act themselves as regularization, reducing the need of problem modification. Based on this property, we suggest to refine successively the discretization of the continuous problem starting from a coarse grid, to find a suitable regularization which renders a good compromise between (measurement) data and regularization error in the estimate. In particular, our experience supports the conjecture, that non-equidistant discretization grids offer advantages over uniform grids.

Key Words: dynamic optimization, adaptive discretization, wavelets, dynamic data reconciliation, state estimation, input estimation, inverse problems.

1 Introduction

Usually process measurements do not reveal complete and correct information about the process state. This is due to the limited number of process quantities accessible to measurement instruments, as well as unavoidable measurement errors caused by imperfect instrumentation and signal processing. Measurement information can be upgraded, in principle, by means of incorporating process knowledge in some mathematical model. Stationary Data Reconciliation (SDR) provides a set of reconciled measurements and an estimate of unmeasured process quantities based on a steady-state process model. Since the pioneering work of Kuehn and Davidson [74] SDR

[1]This work has been supported by the "Deutsche Forschungsgemeinschaft" under grant no. MA1188/6. Corresponding author: marquardt@lfpt.rwth-aachen.de

has been studied extensively and has also been applied successfully to large scale processes in industry [158, 145]. Dynamic Data Reconciliation (DDR) is a natural extension of SDR to dynamic processes and provides a set of reconciled estimates on some time horizon. DDR combines state, parameter and unknown input estimation, and is typically formulated as a dynamic optimization problem restricted by a process model. Interest in DDR was initiated by Almasy [7], who described a data reconciliation technique for linear balance equations. Later on, several authors suggested to solve the problem using nonlinear programming methods in combination with moving horizon techniques (see e.g. Liebmann *et al.* [123]; Ramamurthi *et al.* [154]; Albuquerque and Biegler [4]). DDR might be considered as a generalization of moving horizon state estimation schemes, where some stability and convergence results are available [138, 157, 155]. Although DDR has received much attention recently, it seems that the combination of problem formulation and its associated numerical solution still leaves quite some room for further research to improve understanding and to develop high performance numerical solution strategies.

In this paper, we want to deepen the understanding of such a combination. In particular the properties of dynamic data reconciliation are illuminated, for a simple linear example on a fixed horizon, from the viewpoint of the mathematical theory of *inverse problems* [117]. An application of this theory to the estimation of reaction rates has recently been presented by Mhamdi and Marquardt [136] where the problem is ana
-lysed in the frequency domain after a suitable model transformation. DDR problems are in general ill-posed and require some form of regularization. Regularization stabilizes the numerical solution and aims at simultaneously minimizing the data and regularization errors in the estimate. Regularization in the context of SDR has been recently addressed by Kelly [114]. In this approach, regularization is used to improve numerical stability when solving ill-conditioned algebraic equation systems. Such ill-conditioned systems might arise when unmeasured variables are present, which are hardly observable through the measured variables (see Romagnoli and Sanchez [158], Narasimhan and Jordache [145], for observability in SDR). Although the underlying conceptual idea of regularization is the same for SDR and DDR problems, they differ in one fundamental aspect. SDR is a problem of finite dimension whereas DDR problems involve a differential operator and are therefore of infinite dimension. With the exception of Binder *et al.* [20], who suggested a general DDR problem formulation based on the theory of ill-posed problems, regularization is not explicitly addressed in literature on DDR but always applied implicitly through either discretization of the continuous problem or by starting off with a discrete formulation, since any appropriate discretization of a certain operator itself typically acts as regularization [117, 77]. For example, Liebmann *et al.* [123] and Ramamurthi *et al.* [154] use a coarse discretization of the input variables to obtain smooth estimates as well as to resolve stability problems of the estimation algorithm. The stepsize of the discretization scheme in the continuous case or the sampling time in

the discrete case are of crucial importance to the estimation results and should be chosen judiciously to achieve a good compromise between data and regularization error. However, it is not obvious how to determine a suitable discretization mesh. Here, we suggest a method which automatically generates sequences of discretization grids based on upgrades of the approximation spaces until a good compromise between data and regularization error in the estimates is accomplished. Uniform as well as non-uniform problem adapted upgrades of the approximation spaces are employed. In particular, it is emphasized that non-uniform discretization might lead to improved estimates. The method is built upon a deterministic framework and thus does not depend on statistical assumptions. The results presented apply in principle to a number of suitable discretization methods. However, the discretization employed in this work is based on wavelets, since they allow tailored problem approximations of adjustable accuracy in a natural way.

The article is organized as follows. In Section 2 we introduce the problem considered in the paper and elaborate on its properties by means of an introductory example. In particular, we argue that the problem belongs to the class of inverse problems. Regularization approaches and the associated parameter selection strategies are discussed in Section 3. Here, special emphasis is put on regularization by projection. In Section 4 we introduce the concrete choice of a discretization scheme needed to approximate the continuously formulated problem. The findings are then illustrated using a simple but realistic example (Section 5).

2 Inverse Problems

2.1 The Problem

DDR problems are often formulated as optimization problems restricted to a dynamic model. In this paper we consider a very simple linear scalar problem, since it provides insight and exhibits, at the same time, the essential features of an inverse problem. It is assumed that the time domain of interest has been transformed onto the unity interval $I = [0, 1]$. The DDR problem with a least-squares cost functional has the form

$$\min_{z_0,w} \| z^\delta(t) - z(t) \| \tag{69}$$

$$\text{s.t.} \quad \dot{z}(t) = az(t) + u(t) \qquad +w(t), \quad z(t_0) = z_0. \tag{70}$$

The constraint (70) represents the process model where a is a time invariant known parameter; z and z_0 denote the unknown state and its initial condition; u, w are known and unknown inputs, respectively. $z^\delta(t)$, $t \in I$ denotes a measurement function which is based upon a set of discrete measurement samples $z^\delta(t_i)$ of the state z usually taken from the process and typically corrupted by measurement error indicated by the superscript δ. Note, that no penalty term is present in (69).

2.2 The Properties

The estimation problem belongs to the class of inverse problems, that always searches for causes of observed effects [77]. Typically, inverse problems which involve an integral operator are not well-posed in the sense of Hadamard's definition [117]. Besides the existence of a unique solution, well-posedness requires continuous dependence of the estimate on the measurement data. Suppose, we measure the nominal solution of (70), $\tilde{z}$, for some given initial condition $\tilde{z}_0$ and nominal inputs $\tilde{u}, \tilde{w}$ such that

$$\tilde{w} = \dot{\tilde{z}} - a\tilde{z} - \tilde{u}.$$

Furthermore suppose, that there is alternatively a corrupted measurement function with fixed parameters δ and error frequency n/δ given by

$$z^\delta(t) = \tilde{z}(t) + \delta \sin \frac{n}{\delta} t.$$

Since z^δ is continuously differentiable, the solution to problem (69),(70) is

$$w^\delta = \dot{z}^\delta - a z^\delta - \tilde{u}.$$

It is well known that differentiation of z^δ can cause arbitrarily large errors in the estimation of w for arbitrary small data errors, since

$$\| \tilde{z} - z^\delta \|_\infty = \delta$$

but

$$\| \dot{\tilde{z}} - \dot{z}^\delta \|_\infty = n$$

and

$$\| \tilde{w} - w^\delta \|_\infty = \| n \cos \frac{n}{\delta} t + a\delta \sin \frac{n}{\delta} t \|_\infty .$$

Hence, the errors in $\dot{z}^\delta, w^\delta$ are governed not only by the presumably small error level δ but also by the probably high frequency $\frac{n}{\delta}$. The error in the estimate clearly does not depend continuously on the data and therefore problem (69),(70) is ill-posed in the sense of Hadamard. Note, that this phenomenon is not restricted to (70), but similarly occurs for more general differential equation constraints. It is not due to the special choice of the infinity norm, but holds for general L_p norms, although continuous dependence could be enforced by measuring the data error in a stronger norm containing first or higher order derivatives. However, in practical situations one often prefers to call an error small if the error in the function values is small and therefore some L_p norm is the natural choice. The question immediately arises of what can be done to stabilize the solution, i.e. to reduce the high sensitivity of the estimate with respect to small corruptions of the measurement data.

3 Regularization

Problem (69), (70) can be generalized for subsequent analysis by replacing (70) by the linear operator equation

$$\boldsymbol{T}y \;=\; z,$$

where $y := (w, z_0)$ and $\boldsymbol{T}$ is a bounded linear operator with domain $D(\boldsymbol{T}) \subset Y$, $y \in Y, z \in Z$. In particular, for the process model (70) the linear operator $\boldsymbol{T}$ is given by

$$\boldsymbol{T} : \;(w, z_0) \to z$$

where

$$z(t) \;=\; e^{at} z_0 + \int_{t_0}^{t} e^{a(t-\tau)} (u(\tau) + w(\tau)) d\tau.$$

Usually the corrupted measurement z^δ does not belong to the range $R(\boldsymbol{T})$ since typically no $y = (w, z_0)$ exists such that

$$\boldsymbol{T}y \;=\; z^\delta. \tag{71}$$

In this case the solution of (71) is understood to be the *best approximate solution* denoted by $y^\dagger$ which satisfies (71) in a least-squares sense and at the same time is of minimal norm. The associated operator $\boldsymbol{T}^\dagger$ which maps z^δ to $y^\dagger$, i. e.

$$\boldsymbol{T}^\dagger z^\delta = y^\dagger, \tag{72}$$

is the generalized inverse of $\boldsymbol{T}$ and reflects problem (1),(2). The inverse $\boldsymbol{T}^\dagger$ is generally unbounded, so that stability in the sense of Hadamard is not guaranteed. Hence, even small errors in the measurements z^δ can lead to arbitrarily large errors in the estimate $y^\dagger$. Regularization can be used to stabilize the best approximate solution by introducing a family of bounded transformations $\boldsymbol{T}_r^\dagger$ in (72) such that

$$\lim_{r \to 0} \boldsymbol{T}_r^\dagger z^\delta \;=\; \boldsymbol{T}^\dagger z^\delta, \quad \forall z^\delta, \tag{73}$$

where r is called the regularization parameter. Let z denote the exact data. Given an error bound on the measurement data

$$\| z - z^\delta \| \leq \delta$$

and

$$y_r^\delta := \boldsymbol{T}_r^\dagger z^\delta,$$

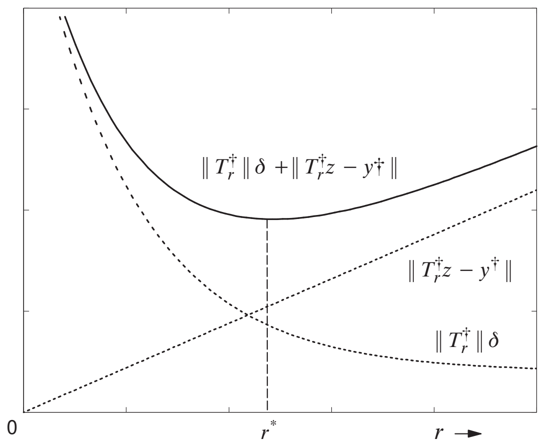


Figure 13: Data and regularization errors (cf. eq. (74).

the following fundamental error statement holds [117, 77]:

$$\| y_r^\delta - y^\dagger \| \quad \leq \quad \| \boldsymbol{T}_r^\dagger \| \delta + \| \boldsymbol{T}_r^\dagger z - y^\dagger \| \,. \tag{74}$$

The bound of the error between the best approximate and estimated solutions splits into two parts. The first term on the right refers to the (measurement) *data error*, while the second denotes the *regularization* or *approximation error* which would be obtained for exact data. The data and regularization errors might be associated with *variance* and *bias* as defined in a *stochastic* setting.[2] The data error largely corresponds to the variance of the estimate but may contain some bias also. In contrast, the regularization error can almost solely be attributed to the bias of the estimate. Since we do not use here any stochastic model and since the interpretation of the terms as either variance or bias would neither in a stochastic setting be straightforward, we prefer to refer to these terms as data and regularization error, respectively. Figure 13 illustrates the general behaviour of the data and regularization errors. By property (73) an increase of r bounds $\boldsymbol{T}_r^\dagger$ and improves stability such that for a constant error level δ the data error is decreased. On the other hand, the increasing difference between $\boldsymbol{T}_r^\dagger$ and $\boldsymbol{T}^\dagger$ causes a growing approximation error. Opposite behavior is observed for decreasing r. Naturally, we are interested in obtaining sharp estimates for $\| y_r^\delta - y^\dagger \|$. Therefore we seek an $r^\star$ which minimizes the sum of data and regularization errors in (74). Parameter selection strategies to find $r^\star$ or a suitable approximation are discussed next.

[2]Regularization explicitly allows bias in the estimate to reduce variance and therefore is conceptually different from the Kalman-Bucy filter, which yields only for linear zero mean processes and white noise with Gaussian statistics bias free estimates with minimal variance [85]. In the general case, when nonlinearities, non-zero mean disturbances or colored noise with unknown statistics are involved, bias free estimates are not guaranteed anymore.

3.1 Parameter selection

Parameter selection strategies are classified in *a-priori* and *a-posteriori* methods [77]. The former refers to the situation where r is chosen before the regularized solution is computed. This is in contrast to the latter where the selection of r is based on the results of the numerical regularization algorithm. Optimal values $r^\star$ might be accessible a-priori for a variety of $\boldsymbol{T}_r^\dagger$ for *known* noise level δ and *known* smoothness of the solution $y^\dagger$, although possibly being numerically expensive. Smoothness assumptions made on $y^\dagger$ are generally of the form

$$y^\dagger = (\boldsymbol{T}^*\boldsymbol{T})^p v, \quad p > 0, \ \| \, v \, \| \leq \rho,$$

where $\boldsymbol{T}^*$ is the adjoint of $\boldsymbol{T}$. The power p can be roughly interpreted as the number of weak derivatives bounded in some norm while ρ is a rough measure of the magnitude of the p-th generalized derivative v of $y^\dagger$ [146]. Unfortunately, a-priori strategies rely on explicit knowledge of δ, p, ρ. While the data error level δ might be accessible in some applications, the degree of smoothness p and the bound ρ of the exact solution $y^\dagger$ are in general at least difficult to estimate. Therefore, a-priori parameter selection strategies are clearly not applicable for most practical problems but are important for the theoretical examination on a specific regularization method.

A-posteriori strategies, which do not require the knowledge of p and ρ for selecting r are more powerful in practice. The best-known a-posteriori strategy is based on *Morozov's discrepancy principle* [141]. The discrepancy principle expresses the fact that the best case to ask for is a residual in the order of δ since smaller residuals are not reliable anyway due to the noise of magnitude δ in the data. Moreover, a smaller regularization parameter destabilizes the solution. Therefore the regularization parameter $r^\star$ is chosen such that the residual (or *discrepancy*) $\| \, \boldsymbol{T} y^\delta_{r^\star} - z^\delta \, \|$ equals the *assumed* data error bound $\hat{\delta}$ including a safeguard factor $\tau \geq 1$, i.e.

$$\| \, \boldsymbol{T} y^\delta_{r^\star} - z^\delta \, \| = \tau \hat{\delta}. \tag{75}$$

Typically $r^\star$ is not determined by solving (75) directly. In practice, one selects a decreasing sequence $\{r_k\}, r_k \to 0$ and sequentially evaluates (75) until $\| \, \boldsymbol{T} y^\delta_{r^\star} - z^\delta \, \| \approx \tau \hat{\delta}$.

In the absence of reliable knowledge on the data error level δ it is necessary to consider alternative (a-posteriori) parameter choice rules that avoid knowledge of the noise level δ and determine a suitable regularization parameter r on the basis of the actual performance of the regularization method under consideration. Such a criterion is given by the L-curve method [93] which is based on an inspection of the residual norm of the computed approximation by relating it to the (semi-) norm of the approximation itself. This is done by plotting $\| \, \boldsymbol{L} \, \| \, y^\delta_r$ versus $\| \, z^\delta - \boldsymbol{T} y^\delta_r \, \|$ in log-log scale for a large range of a decreasing sequence $\{r_k\}, r_k \to 0$. Generally, $\boldsymbol{L}$ is either the identity or a derivative operator. The curve typically has an L-shape with

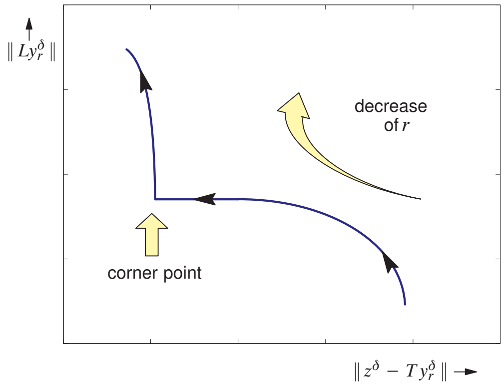


Figure 14: Typical L-curve shape.

a distinct corner point. It is shown qualitatively in Figure 14. There is evidence and substantial practical experience that the corner point reflects a good compromise between data and regularization error in the estimate. Note, that the residual norm in the L-curve criterion as well as in Morozov's discrepancy principle is the same such that the latter is simply a vertical line at $\tau\delta$ in the L-curve plot.

3.2 Direct and indirect methods

A wide variety of regularization methods are available which are typically classified as *direct* and *indirect* approaches. For example, *Tikhonov regularization* [174] is a direct method frequently applied in engineering which minimizes the functional

$$\| \boldsymbol{T}y - z \|^2 + \alpha \| y - y_{ref} \|^2 \tag{76}$$

where α is the regularization parameter, i.e. $r \sim \alpha$. y_{ref} denotes an a priori guess of the solution and ideally is $y^\dagger$. This approach is also known as ridge regression [89] in the context of stabilization of finite dimensional statistical problems. The regularization operator of (76) is given by

$$\boldsymbol{T}_\alpha^\dagger = (\alpha \boldsymbol{I} + \boldsymbol{T}^* \boldsymbol{T})^{-1} \boldsymbol{T}^* \tag{77}$$

with $\boldsymbol{T}^*$ being the adjoint of $\boldsymbol{T}$. The eigenvalues of $\boldsymbol{T}$ tend to zero for ill-posed problems but the eigenvalues of $\alpha \boldsymbol{I} + \boldsymbol{T}^* \boldsymbol{T}$ are bounded away from zero for $\alpha > 0$. Moreover, it can be shown for compact operators $\boldsymbol{T}$ that $\| \boldsymbol{T}_\alpha^\dagger \| \leq \frac{1}{2\sqrt{\alpha}}$ [117]. Often the L_2-norm is chosen as penalty in the Tikhonov functional (76) but it is also common to consider stronger norms (or only semi-norms). In practice, it is nontrivial to find a suitable guess y_{ref} and a wrong guess might lead to unsatisfactory results since the solution is biased.

Regularization is also achieved through indirect methods which rely on the self-regularization property of the underlying solution method. Many iterative methods exhibit such a self-regularization property in the sense that early termination of the iterative process has a regularizing effect [117]. A famous example is *Landwebers iteration formula* [77]

$$\boldsymbol{y}^{m+1} = (\boldsymbol{I} - \beta\boldsymbol{T}^*\boldsymbol{T})\boldsymbol{y}^m + \beta\boldsymbol{T}^* z^\delta,$$

where β denotes a damping factor. The iteration index m is the (hidden) regularization parameter $r \sim \frac{1}{m}$. As $m \to \infty$ the regularization error decreases while the data error increases. The stopping criterion is of crucial importance for the actual performance of the method. The iteration scheme can be interpreted as the steepest decent algorithm applied to the residual norm $\| \boldsymbol{T}y - z \|$. Other iterative methods yielding faster convergence rates such as the CG-method, can be applied as well [117, 92]. *Regularization by projection*, upon which we will focus next in more detail, also reveals such a self regularization property [151].

3.3 Regularization by Projection

Regularization can also be achieved indirectly by projection of (71) into finite dimensional subspaces $Z_0 \subset Z_1 \subset \ldots \subset Z$, $Y_0 \subset Y_1 \subset \ldots \subset Y$. Let $\boldsymbol{Q}_\ell : Z \to Z_\ell$, $\boldsymbol{P}_\ell : Y \to Y_\ell$ be suitable projectors onto Z_ℓ, Y_ℓ, then

$$\boldsymbol{Q}_\ell \boldsymbol{T} \boldsymbol{P}_\ell y = \boldsymbol{Q}_\ell z$$

reduces to a Petrov-Galerkin scheme which represents the discretized problem. The index ℓ might be associated with a stepsize $h \sim \frac{1}{\ell}$ for uniform subspaces which depends on the interval length and projector type. For $\ell \to \infty$ the stepsize becomes increasingly smaller, and hence $h \to 0$. Let us define

$$y_\ell^\dagger := \boldsymbol{T}_\ell^\dagger z_\ell$$

as the least-squares solution of minimal norm to

$$\boldsymbol{T}_\ell y_\ell = z_\ell$$

where

$$\boldsymbol{T}_\ell := \boldsymbol{Q}_\ell \boldsymbol{T} \boldsymbol{P}_\ell, \quad z_\ell := \boldsymbol{Q}_\ell z, \quad y_\ell := \boldsymbol{P}_\ell y.$$

Then, given the error bound $\| \boldsymbol{Q}_\ell(z - z^\delta) \| \le \delta$, it can be shown [77] that

$$\begin{aligned} \| y_\ell^\delta - y^\dagger \| &\le \| \boldsymbol{T}_\ell^\dagger \| \delta + \| \boldsymbol{T}_\ell^\dagger z_\ell - y^\dagger \| \\ &\le \frac{\delta}{\sigma_\ell} + \| y_\ell^\dagger - y^\dagger \|, \end{aligned} \tag{78}$$

where σ_ℓ is the smallest singular value of $\boldsymbol{T}_\ell$. Equation (78) is of the same structure as (74). The first term on the right hand side refers to the data error while the second is the regularization error. For projection methods no regularization parameter is introduced explicitly. However, there is a *hidden* regularization parameter associated to the finite dimensional approximation spaces which affects the smallest singular value of $\boldsymbol{T}_\ell$. The subscript ℓ takes the role of the regularization parameter $r \sim \frac{1}{\ell}$. Due to the spectral properties of continuous operators $\sigma_\ell \to 0$ as $\ell \to \infty$. This suggests to keep the problem discretization rather coarse to bound the data error. On the other hand a coarse problem discretization increases the regularization error. Again, the aforementioned compromise is needed. The question how to choose appropriate finite dimensional spaces to obtain a good compromise is of central importance. Clearly there are infinitely many choices to select the approximation spaces appropriately such that a rigorous search is impractical. Obviously, a procedure is needed which varies the approximation spaces in a systematic way to find the discretization with a suitably low bound in (78). Such a procedure can be obtained for example by application of a refinement approach suggested previously by the authors [20]. The principle idea is to successively *upgrade* the approximation spaces starting from a *first coarse guess* ($\ell = 0$). In each upgrade *new* basis functions are added such that the approximation $\ell+1$ is *refined* either globally or locally. This concept yields sequences of finite dimensional optimization problems of increasing complexity which have to be solved in each refinement step ℓ. The stopping criterion is of particular importance since the successive refinement acts more and more like differentiation as the smallest singular value decreases. This limiting behavior is known to exist also for Luenberger observers where increasingly high observer gains move the smallest eigenvalue of the observer towards $-\infty$ to result in a differentiator [191]. Once a suitable refinement strategy is chosen, the a-priori and a-posteriori parameter selection strategies introduced in Section 3.1 might be applied to find the discretization with a suitably low bound in (78) (for a general discussion on their convergence properties see Plato and Vainikko [151]). In this paper the stopping criterion applied is based on the aforementioned L-curve method since it does not rely on assumptions on the noise level δ. Each approximation ℓ is represented by one point in the L-curve. As soon as a corner point is detected, the refinement can be terminated.

3.4 Combined Regularization

Equation (78) shows that for noisy data and severely ill-posed problems the dimension of the subspace has to be rather low to keep the total error small, since for those problems the smallest $\sigma_\ell(\boldsymbol{T}_\ell)$ decreases rapidly as ℓ increases. To be able to use larger dimensions the projection method might be combined with an additional regularization method. If Tikhonov's method is chosen then the functional

$$\| \boldsymbol{T}_\ell y_\ell - z_\ell \|^2 + \alpha \| y_\ell - y_{\ell,ref} \|^2 \tag{79}$$

has to be minimized. The solution of (79) is given by

$$\boldsymbol{y}^\delta_{\alpha,\ell} = \boldsymbol{T}^\dagger_{\alpha,\ell} \boldsymbol{z}^\delta_\ell,$$

where $\boldsymbol{T}^{\dagger}_{\alpha,\ell} = (\alpha \boldsymbol{I}_\ell + \boldsymbol{T}^*_\ell \boldsymbol{T}_\ell)^{-1}\boldsymbol{T}^*_\ell$. The generalized inverse $\boldsymbol{T}^{\dagger}_{\alpha}$ in (77) is bounded by $\| \boldsymbol{T}^{\dagger}_{\alpha} \| \leq \frac{1}{2\sqrt{\alpha}}$ which must also hold for the projected system independent of ℓ, i. e.

$$\| \boldsymbol{T}^{\dagger}_{\alpha,\ell} \| \leq \frac{1}{2\sqrt{\alpha}}. \tag{80}$$

On the other hand we know that

$$\| \boldsymbol{T}^{\dagger}_{\alpha,\ell} \| \leq \| \boldsymbol{T}^{\dagger}_{0,\ell} \| \leq \frac{1}{\sigma_\ell} \quad \forall\, \alpha \geq 0. \tag{81}$$

Both inequalities (80), (81) have to be satisfied simultaneously such that

$$\| \boldsymbol{T}^{\dagger}_{\alpha,\ell} \| \leq \min\left(\frac{1}{2\sqrt{\alpha}}, \frac{1}{\sigma_\ell}\right).$$

Therefore both, ℓ and α, bound $\| \boldsymbol{T}^{\dagger}_{\alpha,\ell} \|$. If ℓ is chosen such that $\sigma_\ell > 2\sqrt{\alpha}$ then the discretization is the leading regularizer and the solution is hardly affected by the choice of α. Conversely when $2\sqrt{\alpha} > \sigma_\ell$, $\boldsymbol{T}_{\alpha,\ell}$ is stabilized mainly by α and the discretization can be refined arbitrarily provided that α is sufficiently large. One should keep in mind that α is always acting uniformly on the solution in the domain $I = [0, 1]$. On the contrary, a specific discretization might resolve the problem in some areas in greater detail. Therefore α can be considered as a *global* regularizer whereas ℓ can regularize *globally* as well as *locally* depending on the concrete choice of the approximation spaces. If α dominates, then the solution is affected globally *even if* the discretization resolves the problem locally rather than globally.

4 Discretization

So far, the concrete choice of a discretization scheme has not been addressed. The use of wavelets in conjunction with ill-posed problems has been recently explored by a number of authors (Donoho [71]; Dicken and Maass [70]; Freeden and Schneider [81]). Donoho as well as Dicken *et al.* elaborate on the efficiency of the convergence rates using adaptive strategies based on wavelet-vaguelette decompositions. This special class of discretization enables a particularly fast algorithm which can, however, only be applied to a limited class of operators. The advantages of locally adapted regularization parameters for three-dimensional satellite gradiometry problems are recognized by Freeden and Schneider. In line with some of the presented arguments, we suggest the use of wavelets to span the approximation spaces since they allow tailored problem approximations to resolve global as well as local trends. In particular, two methods are presented which refine the problem approximation uniformly as well as non-uniformly. First, some fundamental properties of wavelets are sketched (see e.g. Dahmen [56] for a recent survey and more details).

4.1 Wavelets

The common starting point for the construction of *wavelets* on an *interval* (see e.g. Dahmen *et al.* [56]), which is the relevant setting in the present context, are multiresolution sequences of nested spaces

$$S_{j_0} \subset \ldots \subset S_j \subset S_{j+1} \subset \ldots$$

whose union is dense in $(L_2)(I)$. Here j_0 stands for some coarsest discretization scale. The spaces S_j are usually spanned by compactly supported functions $\varphi_{j,k}$, $j \in \mathcal{I}_j$, like B-splines, whose supports have lengths proportional to 2^{-j}. Thus 2^{-j} can be viewed as the *mesh size* for the space S_j. Given such a multiresolution sequence, one way of updating an approximation $f_j \in S_j$ to a function f is to add some *detail information*

$$f_j + w_j = f_{j+1}$$

where w_j belongs to some *complement* W_j of the space S_j in the next finer space, i.e., $S_{j+1} = S_j \oplus W_j$. With a basis $\{\psi_{j,k}; k \in \mathcal{J}_j\}$ for W_j the detail w_j can be written as

$$w_j = \sum_{k \in \mathcal{J}_j} d_{j,k} \psi_{j,k}.$$

Note that $\psi_{j,k}$ is a linear combination of the $\varphi_{j+1,l}$. With successive decomposition of S_{j+1} into

$$S_{j+1} = S_{j_0} \oplus_{k=j_0}^{j} W_k$$

the function f_{j+1} is associated with a *multiscale representation*

$$f_{j+1} = \sum_{k \in \mathcal{I}_{j_0}} d_{j_0-1,k} \varphi_{j_0,k} + \sum_{l=j_0}^{j} \sum_{k \in \mathcal{J}_l} d_{l,k} \psi_{l,k}.$$

Under certain assumptions on the complements W_j the complement basis functions $\psi_{j,k}$ are called *wavelets*. The whole collection

$$\{\varphi_{j_0,k} : k \in \mathcal{I}_{j_0}\} \cup \{\psi_{j,k} : k \in \mathcal{J}_j, j = j_0, j_{0+1}, \ldots\}$$

is a *basis* for $L_2(I)$. Wavelets are constructed such that the Euclidean norm of the wavelet coefficients of f is always proportional to the L_2 norm of f. Thus discarding small coefficients $d_{j,k}$ will cause only small changes in approximate representations of f. Moreover, it can be shown that the values $|d_{j,k}|$ will be quite small when f is smooth on the support of $\tilde{\psi}_{j,k}$. Therefore, functions that exhibit only strong variations locally, can be approximated very accurately by linear combinations of relatively few wavelet basis functions corresponding to the *significant* coefficients $d_{j,k}$. Denoting by Λ the set of the corresponding indices (j,k), the finite basis

$$\Psi_\Lambda := \{\psi_{j,k} : (j,k) \in \Lambda\}$$

is expected to lead to a particularly *adapted* approximation to f. We refer to Ψ_Λ as a *sparse* wavelet basis in contrast to a *full* wavelet basis Ψ_{Λ^J}, corresponding to index sets

$$\Lambda^J := \{(j,k) : j_0 - 1 \leq j \leq J - 1, k \in \mathcal{J}_j\}$$

and obtained by cutting off all frequencies above scale J. The former may be associated with a possibly highly *nonuniform adapted mesh* while the latter induces a *uniform* discretization of mesh size 2^{-J}.

4.2 Uniform and adaptive discretization

In our realizations the approximation spaces Z_ℓ, Y_ℓ introduced above are spanned by a wavelet basis. Specifically, the approximation for the states z and measurements z^δ are generated by piecewise linear wavelets. The inputs w, u are expanded in terms of piecewise constant (Haar) wavelets. Starting from a uniform coarse grid $\ell = 0$ the upgrades of the approximation spaces can be either accomplished uniformly or non-uniformly. For uniform adaptation, in each upgrade the complete next scale is added to the approximation, e.g. the basis is $\Psi^\ell_{\Lambda^J}$ with $J := j_0 + \ell$. Alternatively, a sparse basis can also be applied since it might lead to more economical discretizations. Here our own method has been employed [23] to generate sparse discretization grids. The basic idea followed in this approach is to successively identify trial functions with large directional sensitivities and thus use them for an improved problem approximation. As a simplification we keep in this work the spaces of z, z^δ constant at a very fine resolution and vary only the approximation spaces for the input w. Based on the computed estimates at the refinement stage ℓ, candidate sets of potential basis functions for the upcoming upgrade $\ell + 1$ are specified which are not yet used in the current approximation. The adaptive refinement of w is based on a sensitivity evaluation of the Lagrange function with respect to the functions contained in the candidate sets since it can be expected that large absolute values indicate substantial impact on the reduction of the cost function. Basis functions which are associated to large sensitivities are used for an upgraded approximation at refinement stage $\ell + 1$. As candidate sets, simply the local neighbors are taken with respect to scale and translation of the current approximation.

5 Example

Our example is built upon calorimetric estimation [165] applied to a nonisothermal CSTR where a first order reaction $A \to B$ takes place. The nonlinear model of the

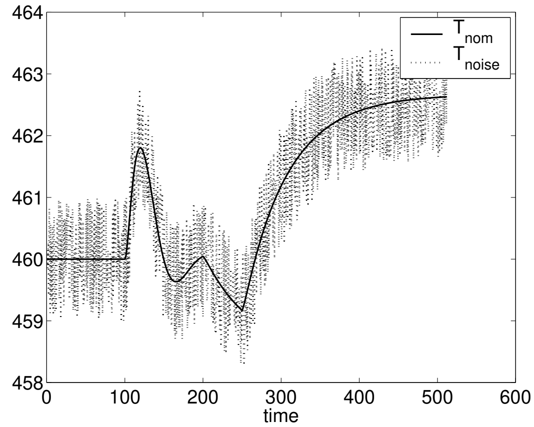


Figure 15: Nominal and corrupted temperatures

reactor and a temperature controller is simulated rigorously assuming a disturbance in the feed concentration and step changes in the inlet temperature. Given noisy temperature measurements as well as the transients of inlet and cooling temperature T_0 and T_{cool}, the task is to find reconciled values of the heat of reaction Q_{rea} without knowledge of the reaction kinetics. The estimation model (70) is here the energy balance equation

$$\dot{T} = \frac{q}{V}(T_0 - T) + Q_{rea} - \frac{UA}{(\rho c_p V)}(T - T_{cool}) \tag{82}$$

with flow rate $q = 10\ cm^3/s$, vessel volume $V = 10^3\ cm^3$, heat transfer coefficient $U = 5\ 10^{-4}\ cal/(cm^2 sK)$, heat transfer area 10 cm^2, liquid density $\rho = 0.001\ g/cm^3$, and heat capacity $c_p = 1\ cal/(gK)$. The nominal solution of (82) denoted by T_{nom} is corrupted by uniformly distributed noise of magnitude 1 to be used as simulated perturbed measurements T_{noise}. T_{nom}, T_{noise} are shown in Figure 15. The reconciliation problem is of least-squares form and given by

$$\min_{T(0),Q_{rea}} \int_{t=0s}^{512s} (T_{noise} - T)^2 d\tau \tag{83}$$

subject to (82). The formulation matches the general problem given by (69), (70) and is discretized as outlined before. The approximation spaces for T_{nom}, T_{noise} are kept constant at scale $J = 9$ while we employ uniform as well as sparse wavelet bases $\Psi^{\ell}_{\Lambda^J}$ and Ψ^{ℓ}_{Λ} to approximate Q_{rea}. The refinement is started from an initial uniform mesh $\ell = 0$ at the approximation scale $J = 1$ and is then gradually upgraded. Estimates obtained employing a uniform basis at the refinement steps $\ell \in \{3, 4, 5\}$ are shown in Figure 16, which corresponds to approximation scale $J \in \{4, 5, 6\}$. The observed differentiation effect due to a decreasing smallest singular value σ_ℓ can be seen clearly. The L_2-error of $Q_{rea,estim}$ and $Q_{rea,nom}$ vs. refinement index ℓ are shown

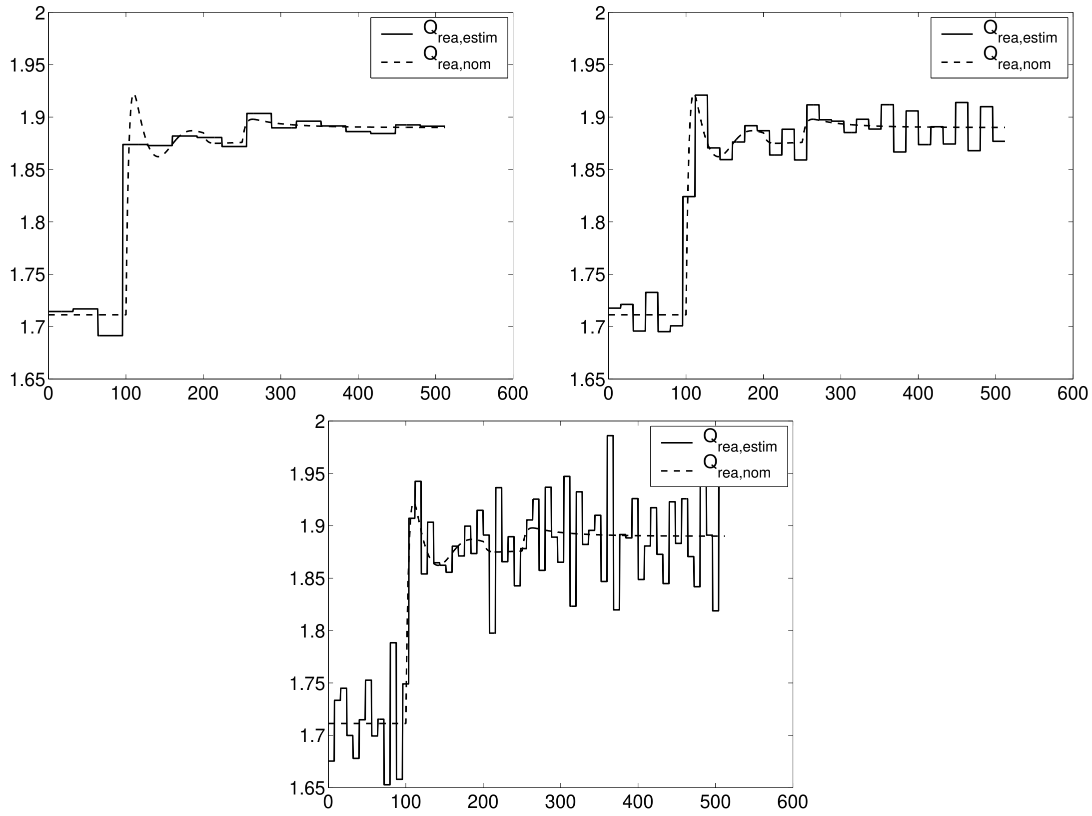


Figure 16: $Q_{rea,estim}$ on uniform mesh: $\ell = 3,\ J = 4$ (top); $\ell = 4,\ J = 5$ (middle); $\ell = 5,\ J = 6$ (bottom).

for all upgrades in Figure 17. However, in practice, there is no information about the nominal values or the noise level δ. The L-curve in Figure 18 is used instead to detect a good compromise between data and regularization error. The first derivative of the estimate is taken as semi-norm $\| \boldsymbol{L} Q_{rea,estim} \|_{(L_2)}$. As residual norm we use $\| T_{noise,J} - \boldsymbol{T}_J Q^{\ell}_{rea,estim} \|_{(L_2)}$ where the measurements and continuous operator are evaluated at high resolution (scale $J = 9$). Since the example is a simulated experiment where the noise level is available, we can evaluate $\delta = \| T_{nom} - T_{noise} \|_{(L_2)}$ to obtain a sharp L_2-estimate. This value is included in the L-curve plot for $\tau = 1$ with Morozov's method at $\hat{\hat{\delta}} = \delta = 0.47$ as a straight line. According to Morozov's discrepancy principle $\tau\delta$ is the value which should be achieved as residual error norm by any suitable regularization strategy. As one can see in Figure 18, the refinement level $\ell = 4$ associated to the corner point of the uniform discretization coincides quite nicely with Morozov's criterion and with the best achievable estimates taken from Figure 17.

The (L_2) error and the L-curve are shown as well for non-uniform refinement in Figure 17 and Figure 18. In each refinement step only one trial function with

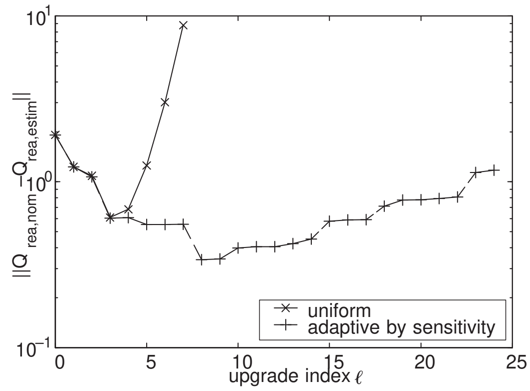


Figure 17: L_2 error of the estimates.

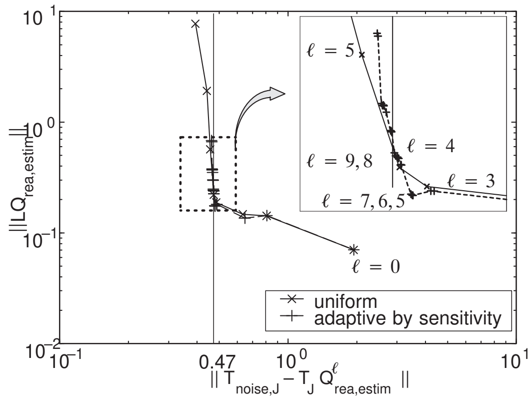


Figure 18: L-curve and Morozov's discrepancy value.

the largest sensitivity has been added, but this can easily be enlarged to obtain faster convergence rates. Again, the corner point of the L-curve coincides well with the minimal (L_2) error. However, a range of discretizations corresponds to the points on the curve near its corner. The (L_2) error reveals that an approximation quality is achieved in the non-uniform case which can not be obtained using uniform discretization. This is also seen by comparing Figure 16 to Figure 19 where $Q^{\ell}_{rea,estim}$ is shown at upgrade step $\ell = 6$ and $\ell = 8$. In such adaptive discretizations the sharp transient can be resolved using small stepsizes while the flat areas are discretized with larger stepsizes and therefore keeps the noise in the estimates low. On the contrary, uniform discretization either resolves the sharp transient at the cost of an increased noise in the flat areas or does not resolve the sharp transient, while the noise in the flat areas are kept low. Note, that for the non-uniform refinement in Figure 19 we can not attribute a uniform approximation scale J, rather we would have to refer to the complete index set Λ_ℓ to characterise the discretization.

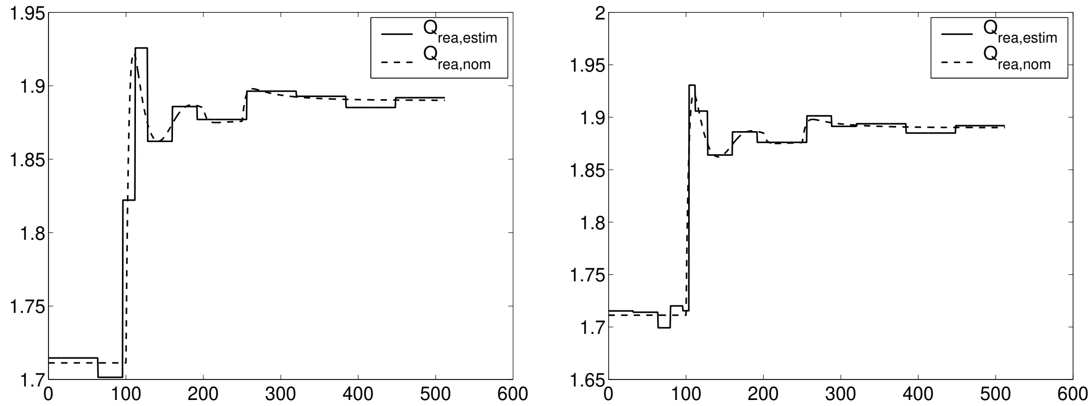


Figure 19: $Q_{rea,estim}$ on adaptive mesh: $\ell = 6$ (top); $\ell = 8$ (bottom).

Next the behaviour is examined if regularization by projection is applied in combination with Tikhonov's method. The cost function (83) is augmented by the penalty

$$\alpha \int_{t=0s}^{512s} (Q_{rea,estim} - Q_{rea,ref})^2 d\tau.$$

$Q_{rea,ref}$ is typically not known but should be guessed as well as possible since a wrong guess might introduce a large bias. As a guess a constant profile $Q_{rea,ref} = 1.8\frac{K}{s}$ is assumed. The reconciliation problem is solved for each combination of $\alpha \in \{0, 5, 25, 50, 75, 100, 250, 500\}$ and uniform meshes at resolution $J \in \{1, 2, 3, 4, 5, 6, 7\}$. Each point in Figure 20 reflects one combination of α and J. The discretizations scales $J = 1, \ldots, 7$ starting from $\ell = 0$ form one curve for a specific choice of α. The curve for $\alpha = 0$ is the same as in Figure 18 for uniform discretization. At $\alpha \sim 50$ a qualitative change in the behaviour occurs. For α & 50 the Tikhonov regularization is dominating and an improved approximation does not affect the stability of $\boldsymbol{T}_{n,\alpha}^{\dagger}$. For large values of α there is clearly an over-regularization and $Q_{rea,estim}$ is dominated by the choice of $Q_{rea,ref}$ due to the penalty α. However, for α . 50 the discretization destabilizes the estimates and there is an increase of the noise in the estimates. These curves reveal the expected L-curve behaviour for the sequence of refined discretizations. Note, that if the combinations J and α are virtually redrawn such that all $\{\alpha_k\}, k = 1, \ldots 8$ for a selected ℓ are connected, then the resulting curves would also reveal L-shape (not shown).

The transient $Q_{rea,estim}$ obtained for $\alpha = 25,\ J = 6$ and $\alpha = 50,\ J = 7$, which are close to the value suggested by Morozov's discrepancy principle, are shown in Figure 21. The noise in the estimates is relatively small but still larger compared to the estimates shown in Figure 19. As mentioned before, the Tikhonov regularization acts uniformly and does not allow a locally adapted regularization. Therefore the fast transient at $t \approx 110s$ is not resolved at an acceptable noise level in the flat areas

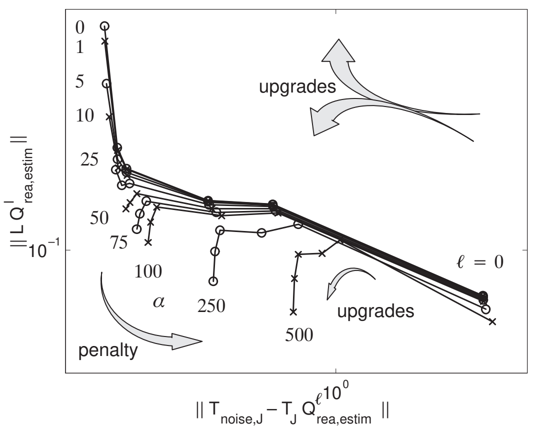


Figure 20: Tikhonov regularization for different uniform discretizations.

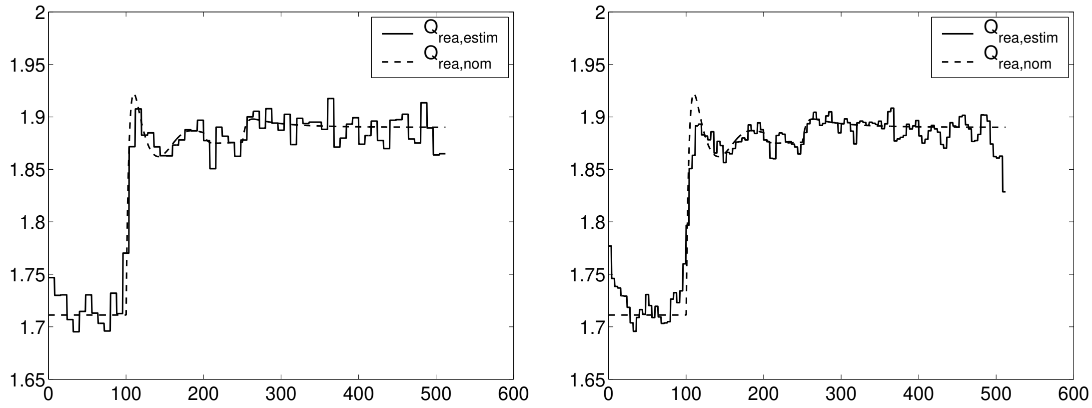


Figure 21: $Q_{rea,estim}$ on uniform mesh: $\alpha = 25$ $J = 6$ (top); $\alpha = 50$ $J = 7$ (bottom).

of the estimate. This is also the case if the problem is discretized non-uniformly (not shown). To illustrate the effect of over-regularization the transient $\alpha = 250,\ J = 7$ is shown in Figure 22 where the estimates are smooth but have a large bias. Note, that the estimates reveal in particular significant bias near the boundaries which is due to the (wrong) guess $Q_{rea,ref} = 1.8\frac{K}{s}$. The bias gets worse penetrating also into the interior region for bad guesses such as $Q_{rea,ref} = 0\frac{K}{s}$. Therefore, a good guess of the solution is crucial for the performance. An improved guess of the reference function $Q_{rea,ref}$ might be obtained by an estimation of Q_{rea} on a suitable (possibly non-uniform) grid with $\alpha = 0$. This estimate can then be used as reference function $Q_{rea,ref} = Q^{\star}_{rea,estim}$ and a higher resolution might be obtained applying Tikhonov's method with an appropriately chosen α.

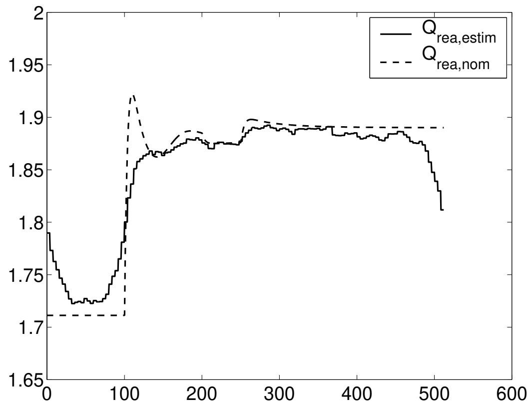


Figure 22: $Q_{rea,estim}$ on uniform mesh: $\alpha = 250,\ J = 7$.

6 Conclusions

DDR has been discussed from the viewpoint of inverse problems. The DDR problem is ill-posed and requires regularization of some form. A well known method is Tikhonov regularization which adds a suitable penalty to the cost function. However, regularization might also be obtained by discretization. Discretization indirectly regularizes the DDR problem and has to be chosen appropriately to minimize data and regularization error in the estimates. Based on successive upgrades a novel method is suggested to automatically generate a hierarchy of discretization grids until a best compromise between data and regularization error is achieved. Problem adapted uniform as well as non-uniform upgrades have been considered. In particular, it turned out for the specific example, that non-uniform approximation leads to estimates that cannot be achieved employing a uniform grid. The regularization framework has been presented for linear models on a fixed time window and serves as a starting point for moving horizon approaches, where a similar behavior is expected. The method should carry over to control problems which are dual to estimation problems. The analysis of ill-posed problems can be also extended to problems with nonlinear models. However, in the nonlinear case the concept of ill-posedness can not be motivated by singular values and therefore the obtained estimates are valid only qualitatively. The regularization approach has been illustrated using a model with one unknown input. The problem of quantitatively assessing the best compromise becomes increasingly difficult for a growing number of unknown input functions due to the combinatorial increase of parameter variations. Here suitable, low-dimensional strategies have to be developed to effectively choose the regularization parameters.

Iterative Algorithms for Multiscale State Estimation, Part 1: Concept[3]

Thomas Binder, Luise Blank, Wolfgang Dahmen, Wolfgang Marquardt

in: Journal of Optimization Theory and Applications (JOTA), Vol. 111 (3), 501-527, 2001.

Abstract

The objective of the present investigation is to explore the potential of multiscale refinement schemes for the numerical solution of dynamic optimization problems arising in connection with chemical process systems monitoring. State estimation is accomplished by the solution of an appropriately posed least–squares problem. To offer at any instance of time an approximate solution a hierarchy of successively refined problems is designed using a wavelet based Galerkin discretization. In order to fully exploit the approximate solution obtained at any stage also for an efficient treatment of the arising linear algebra tasks, we wish to employ iterative solvers. In particular, we will apply a nested iteration scheme to the hierarchy of arising equation systems and adapt the Uzawa algorithm to the present context. Moreover, we will show that using wavelets for the formulation of the problem hierarchy the largest eigenvalues of the resulting linear systems can be effectively controlled with scaled diagonal preconditioning. Finally, we deduce appropriate stopping criteria and illustrate the characteristics of the solver with a numerical example.

Key Words: Dynamic optimization, optimal control, wavelets, nested iteration, iterative linear algebra, preconditioning.

1 Introduction

Monitoring chemical process systems requires repetitive on–line estimation of the entire state, which is not accessible by measurement. Process states and other derived quantities must therefore be determined via model based state estimation employing measured data. State estimation can be formulated as a constrained *optimization*

[3]This work has been supported by the “Deutsche Forschungsgemeinschaft” under grant no. MA1188/6

problem, where the difference between modeled outputs $\boldsymbol{y}$ and measured data $\boldsymbol{z}$ on a receding horizon is to be minimized [137, 144, 157]. Note that $\boldsymbol{z}$ is a continuous representation of the discrete measurements $\boldsymbol{z}(t_k)$ at not necessarily equidistant sampling times t_k. The dynamic behavior of the plant is usually modeled by a large–scale system of differential–algebraic equations to predict in addition to $\boldsymbol{y}$ the states $\boldsymbol{x}$ for given controls $\boldsymbol{u}$. Moreover, additive model correction terms are introduced into the model equations as functions $\boldsymbol{v}$ and $\boldsymbol{w}$ which represent unmodeled phenomena. These functions have to be estimated as well. These model equations and additional inequality constraints, reflecting for example physical bounds on the states, are the constraints in the optimization problem. The introduction of model correction terms actually enhances ill–posedness, i.e. nonuniqueness as well as high sensitivity to inevitable measurement noise may occur [20, 102] and the generalized inverse is discontinuous [29]. In the context of dynamic data reconciliation [20] one major regularization approach is the inclusion of $\boldsymbol{v}$ and $\boldsymbol{w}$ as quadratic penalty terms in the cost functional. In addition, the unknown initial values of the states could be included. This is closely related to Tikhonov regularization [173]. For a more detailed discussion of the form of the objective function we refer to [20].

The *key requirement* is to provide a reliable state estimate *at any instant of time*, since the time required to calculate a solution as well as the time available in a multitasking workstation environment are *not* known beforehand. Further, the estimation problem need to be completed within the cycle time interval of the monitoring system. One faces the following difficulty when employing established solvers based on a fixed a priorly chosen discretization. On one hand, to guarantee the availability of an estimation within an expected time interval one may choose a discretization that will then turn out to be too coarse while the remaining time may not permit a restart with a refined discretization. On the other hand, if the resolution is chosen too fine initially there may be no result at all within the cycle time interval of the estimates.

As an alternative we therefore propose to develop a suitable *hierarchy of optimization problems* with increasing resolution. Thus already after a hopefully very short period of time the coarsest approximate solution can serve as a minimal response. During the remaining time interval this *initial solution* is to be *upgraded* so that the full available time span is exploited in an optimal way. Moreover, for any discretization level the corresponding discrete problem has to be solved only with an accuracy that is comparable to the corresponding discretization error. Hence, in the very spirit of classical *nested iteration*, the current approximation can be exploited as initial guess and the current error has to be only reduced by a fixed factor when progressing to the next discretization level. Consequently, the use of iterative methods suggests itself. The optimization community can nowadays resort to a highly advanced supply of direct solvers for the linear systems of equations that ultimately arise (possibly after linearization) from the optimization problem [18, 34]. However, the above aspect makes us believe that a thorough investigation of *iterative schemes* as solvers for these *linear systems of equations* as an alternative is due in the present context.

Our approach is based on *Galerkin discretizations*. Consequently, the use of hierarchically organized bases for realizing the above concept is quite natural. In particular, we employ *wavelets*. In the context of iterative solvers they are one corner stone for constructing a good preconditioner leading to condition numbers independent of the discretization level. They also facilitate adequate adaptive refinement strategies.

Of course, the development and validation of such a concept is a rather complex task. Our first experiments quickly revealed a multitude of different interfering effects. Therefore we found it necessary to scale the problem down to an extent that those features can be identified. Hence, we will only address here *linear problems* and *suppress inequality constraints*. A successful treatment of this problem class will be essential for handling more realistic problem formulations subsequently.

The objective of this paper is to develop the basic ingredients for the above program. The outline of the paper is as follows: In Section 2 we describe first the model problem in more detail and discuss aspects of the sources of the ill–conditioning. We proceed in Section 3 with our multiscale discretization. In particular, the corresponding Karush–Kuhn–Tucker–system (KKT–system) is given in wavelet coordinates. For the solution of the optimization problem on each discretization level we construct and analyze a simplified Uzawa algorithm exploiting the specific problem structure in Section 4. Conceptionally, this method is the application of the pcg–method to the Schur complement, which is explicitly given in Subsection 4.2. Section 5 addresses the ill–conditioning of the KKT–systems and its specific sources. We shall see that wavelet concepts can be used to control large eigenvalues. In fact, a simple diagonal scaling referred to as (approximate) Jacobi preconditioner will turn out to provide condition numbers bounded independently of the discretization level. In Section 6 we sketch the nested iteration scheme for the solution of the sequence of systems. Moreover, we derive relevant stopping criteria based upon the preconditioned systems. We conclude in this section with formulating the complete algorithm and an application to a simple example. In the last section we briefly summarize the results and draw some conclusions.

In Part 2 [27] of this paper the proposed simplified Uzawa scheme and the Jacobi preconditioner will be applied to test examples with typical obstructions. In particular, the interplay between state normalization, the choice of regularization parameters and preconditioning will be studied .

2 Estimation Problem

Our main objective in this paper is the efficient solution of the systems of discrete equations arising from optimization problems described below. In particular, the solvers should support the multiscale concept outlined in the previous section. Since the same issues would arise after linearization of a nonlinear problem we confine the discussion here to a *linear model problem with equality constraints* only.

The receding horizon will always be scaled to $[0,1]$. Since all functions will therefore refer to this interval we will omit in the sequel explicit references to this domain. Then, for given $\boldsymbol{z} \in (L_2)^{n_y}$ and $\boldsymbol{u} \in (L_2)^{n_u}$ and unknown functions $\boldsymbol{x} \in (H^1)^{n_x}$, $\boldsymbol{y} \in (L_2)^{n_y}$, $\boldsymbol{v} \in (L_2)^{n_v}$ and $\boldsymbol{w} \in (L_2)^{n_w}$ with $n_w \le n_x$, $n_v \le n_y$, the resulting linear optimization problem has the form

$$\min_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{v},\boldsymbol{w}} \int_0^1 \left\{ (\boldsymbol{y}-\boldsymbol{z})^T \boldsymbol{Q} (\boldsymbol{y}-\boldsymbol{z}) + \boldsymbol{v}^T \boldsymbol{R}_v \boldsymbol{v} + \boldsymbol{w}^T \boldsymbol{R}_w \boldsymbol{w} \right\} \, dt \tag{84}$$

subject to the constraints

$$\begin{aligned} \dot{\boldsymbol{x}} - \boldsymbol{A}\boldsymbol{x} - \boldsymbol{W}\boldsymbol{w} &= \boldsymbol{B}\boldsymbol{u}, \qquad t \in [0,1], && (85) \\ \boldsymbol{y} - \boldsymbol{C}\boldsymbol{x} - \boldsymbol{V}\boldsymbol{v} &= \boldsymbol{0}. && (86) \end{aligned}$$

In principle, the weights $\boldsymbol{Q}, \boldsymbol{R}_v$ and $\boldsymbol{R}_w$ could be operators, for example chosen in such a way that the cost functional is equivalent to the square of Sobolev norms. This way, regularity considerations could be elegantly included in the estimation problem. Often $\boldsymbol{Q}$ is preferably chosen though as a constant diagonal matrix reflecting the confidence in the measurements [13]. The right choice of the regularization terms $\boldsymbol{R}_v$ and $\boldsymbol{R}_w$ is still open at least for nonlinear problems (see [157] for a thorough discussion). Therefore, diagonal matrices are usually employed for pragmatic reasons. The matrices $\boldsymbol{V}$ and $\boldsymbol{W}$ are identity matrices possibly extended by zero rows.

For solving the optimization problem, we reformulate the constraints in a weak sense. Here and in the following we denote by $\langle\ ,\ \rangle$ the L_2 scalar product $\langle s,u\rangle = \int_0^1 s(t)u(t)dt$ for $s,u \in L_2$. For convenience we introduce for function vectors $\boldsymbol{s},\boldsymbol{u} \in (L_2)^n$ the scalar product notation $\langle \boldsymbol{s},\boldsymbol{u}\rangle = \int_0^1 \boldsymbol{s}^T(t)\boldsymbol{u}(t)dt$ and the notation $\langle\langle \boldsymbol{s},\boldsymbol{u}\rangle\rangle = \int_0^1 \boldsymbol{s}(t)\boldsymbol{u}^T(t)dt$ for the matrix of all possible scalar products $(\langle s_i,u_j\rangle)_{(i,j)}$. In these terms the constraints take the form

$$\begin{aligned} \langle \boldsymbol{\zeta}_1, \dot{\boldsymbol{x}} - \boldsymbol{A}\boldsymbol{x} - \boldsymbol{W}\boldsymbol{w}\rangle &= \langle \boldsymbol{\zeta}_1, \boldsymbol{B}\boldsymbol{u}\rangle \qquad \text{for all } \boldsymbol{\zeta}_1 \in (L_2)^{n_x} && (87) \\ \langle \boldsymbol{\zeta}_2, \boldsymbol{y} - \boldsymbol{C}\boldsymbol{x} - \boldsymbol{V}\boldsymbol{v}\rangle &= \boldsymbol{0} \qquad \text{for all } \boldsymbol{\zeta}_2 \in (L_2)^{n_y}. && (88) \end{aligned}$$

For convenience we gather the unknowns in $\boldsymbol{\Upsilon} := (\boldsymbol{x}^T, \boldsymbol{y}^T, \boldsymbol{v}^T, \boldsymbol{w}^T)^T \in \Upsilon$ and the test functions in $\boldsymbol{\zeta} := (\boldsymbol{\zeta}_1^T, \boldsymbol{\zeta}_2^T)^T \in \mathcal{M}$ where

$$\Upsilon \ := \ \left(H^1\right)^{n_x} \times (L_2)^{n_y+n_v+n_w}, \quad \mathcal{M} := (L_2)^{n_x+n_y}.$$

Thus denoting by $a(\boldsymbol{\Upsilon}', \boldsymbol{\Upsilon})$ the bilinear form with $\boldsymbol{\Upsilon}, \boldsymbol{\Upsilon}' \in \Upsilon$ resulting from the cost functional in (84) and writing $b(\boldsymbol{\zeta}, \boldsymbol{\Upsilon})$ for the combined bilinear forms resulting from the left hand side of (87), (88), i.e.,

$$\begin{aligned} a(\boldsymbol{\Upsilon}, \boldsymbol{\Upsilon}') &= \langle \boldsymbol{y}, \boldsymbol{Q}\boldsymbol{y}'\rangle + \langle \boldsymbol{v}, \boldsymbol{R}_v \boldsymbol{v}'\rangle + \langle \boldsymbol{w}, \boldsymbol{R}_w \boldsymbol{w}'\rangle \\ b(\boldsymbol{\zeta}, \boldsymbol{\Upsilon}) &= \langle \boldsymbol{\zeta}_1, \dot{\boldsymbol{x}} - \boldsymbol{A}\boldsymbol{x} - \boldsymbol{W}\boldsymbol{w}\rangle + \langle \boldsymbol{\zeta}_2, \boldsymbol{y} - \boldsymbol{C}\boldsymbol{x} - \boldsymbol{V}\boldsymbol{v}\rangle, \end{aligned}$$

the weak formulation of the necessary conditions of the minimization problem (84)–(86) reads as follows: Find $\boldsymbol{\Upsilon} \in \Upsilon$ and the Lagrange multiplier $\boldsymbol{\lambda} = (\boldsymbol{\mu}^T, \boldsymbol{\nu}^T)^T \in \mathcal{M}$ such that

$$a(\boldsymbol{\Upsilon}, \boldsymbol{\Upsilon}') + b(\boldsymbol{\lambda}, \boldsymbol{\Upsilon}') = \langle 2\boldsymbol{Q}\boldsymbol{z}, \boldsymbol{\Upsilon}' \rangle \quad \text{for all } \boldsymbol{\Upsilon}' \in \Upsilon, \tag{89}$$

$$b(\boldsymbol{\zeta}, \boldsymbol{\Upsilon}) = \langle \boldsymbol{\zeta}_1, \boldsymbol{B}\boldsymbol{u} \rangle \quad \text{for all } \boldsymbol{\zeta} \in \mathcal{M}. \tag{90}$$

In view of the Riesz representation theorem the weak formulation (89)–(90) is equivalent to the operator equation

$$\boldsymbol{\mathcal{L}} \begin{pmatrix} \boldsymbol{\Upsilon} \\ \boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} \boldsymbol{h}(\boldsymbol{z}, \boldsymbol{u}) \\ \mathbf{g}(\boldsymbol{z}, \boldsymbol{u}) \end{pmatrix} \tag{91}$$

with unknowns $\boldsymbol{\Upsilon} \in \Upsilon$ and $\boldsymbol{\lambda} \in \mathcal{M}$. Thus in this setting the solution $\boldsymbol{\lambda}$ of the dual optimization problem is automatically part of the solution.

The fact that the problem formulation (84)–(86) is well–posed means that the mapping $\boldsymbol{\mathcal{L}}$ is a topological isomorphism from $\Upsilon \times \mathcal{M}$ to the dual $\Upsilon' \times \mathcal{M}'$, i.e., there exist positive constants $c, \tilde{c}$ such that

$$c\left(\|\boldsymbol{\Upsilon}\|_{\Upsilon}^2 + \|\boldsymbol{\lambda}\|_{\mathcal{M}}^2\right) \leq \|\boldsymbol{\mathcal{L}}\begin{pmatrix} \boldsymbol{\Upsilon} \\ \boldsymbol{\lambda} \end{pmatrix}\|^2_{((H^1)^{n_x})' \times (L_2^{n_x+2n_y+n_v+n_w})} \leq \tilde{c}\left(\|\boldsymbol{\Upsilon}\|_{\Upsilon}^2 + \|\boldsymbol{\lambda}\|_{\mathcal{M}}^2\right). \tag{92}$$

Note that some components of the unknowns are now measured in stronger norms than the L_2-norm. Of course, the quotient $\tilde{c}/c$ may be very large, which means, the uniquely solvable, continuous estimation problem will exhibit a *very large condition number*. There are essentially three interfering potential origins for large condition numbers. Firstly, the original problem of state estimation without regularization (i.e. $\boldsymbol{R}_v = \mathbf{0}, \boldsymbol{R}_w = \mathbf{0}$) may be ill–posed and the way of regularization affects the constants in (92). Secondly, the underlying state equations can be very stiff. And last but not least the observability measure of the model equations may be very low. *Observability measures* quantify the relative degree on how well the state $\boldsymbol{x}$ can be reconstructed from the observations. There exist quite a number of different observability measures (see Part 2 and [110]). Which observability measure quantifies best the influence on the condition numbers is not clear yet. Moreover, regularization alters the extent to which the observability measure affects the condition of the optimization problem. Nevertheless, a quantitative understanding of this interplay is still lacking.

As discretization technique we will apply the Galerkin method based on wavelets. The wavelet based Galerkin method is in principle not new, but essentially confined to elliptic boundary value problems and boundary integral equations (see [47, 56] and references therein). In the context of simulation and dynamic optimization of ordinary differential equations several related wavelet approaches appeared recently in the literature [42, 104, 192]. However, to our knowledge wavelet based Galerkin methods for state estimation has so far not been used and studied in the context of nested iterations.

3 Discretization and Resulting KKT–System

In order to discretize the weak formulation (89)–(90) of the optimization problem (84)–(86) with a Galerkin method we have to choose appropriate finite dimensional subspaces Υ_Λ and $\mathcal{M}_\Lambda$ of Υ and $\mathcal{M}$. The idea of the refinement approach presented in the introduction, is based on the use of hierarchically organized bases. In particular wavelets offer the following advantages: (i) they permit a stable representation of data in a multiscale format where the involved coefficients represent *updates* with increasing resolution; (ii) they support preconditioning. For an introduction and overview over the wavelet concepts we refer the reader who is unfamiliar with wavelets to [43, 56, 64].

To ensure possibly sparse matrix patterns we will employ in this paper wavelets with possibly small support and hence minimal regularity. Since only first order derivatives appear in the constraints (87) globally continuous piecewise linear spline wavelets are appropriate for discretizing the functions $\boldsymbol{x}$, $\boldsymbol{y}$, $\boldsymbol{v}$. More precisely, we employ wavelets $\psi_{j,k}$ determined by the dual multiresolution of order 2, which are adapted to the interval $[0,1]$ [59]. Thus both the primal wavelets $\psi_{j,k}$ as well as the dual wavelets $\tilde{\psi}_{j,k}$ have second order vanishing moments. Moreover, collecting for convenience all functions $\psi_{j,k}$, $\tilde{\psi}_{j,k}$ in the arrays $\mathbf{\Psi}$ and $\tilde{\mathbf{\Psi}}$ respectively, they satisfy $\langle\langle \mathbf{\Psi}, \tilde{\mathbf{\Psi}} \rangle\rangle = \mathbf{I}$. Note that the corresponding primal multiresolution is generated by the classical *hat function* $\varphi(x) := 1 - |x|$ for $x \in [-1,1]$ and $\varphi(x) := 0$ elsewhere. For the last component $\boldsymbol{w}$ of the space Υ we choose the *Haar basis* $\mathbf{\Psi}^H$. Thus the generating scaling function is simply $\varphi^H := \chi_{[0,1)}$. Note that since the Haar basis is orthonormal it equals its dual, i.e. $\langle\langle \mathbf{\Psi}^H, \mathbf{\Psi}^H \rangle\rangle = \boldsymbol{I}$. Likewise all components of the first group of Lagrange multipliers $\boldsymbol{\mu}$ are discretized with the aid of the Haar basis $\mathbf{\Psi}^H$ while the dual wavelets in $\tilde{\mathbf{\Psi}}$ are used for the second group of components of the Lagrange multipliers $\boldsymbol{\nu}$ in $\mathcal{M}_\Lambda$. Thus the state equations (85) are tested by Haar wavelets $\psi^H_{j,k}$ and the output equations (86) are tested by wavelets $\tilde{\psi}_{j,k}$ dual to the piecewise linear wavelets $\psi_{j,k}$.

Note that the chosen collections of wavelets form *bases* for the relevant function spaces. Hence we obtain a infinite dimensional but discretized problem formulation which is still equivalent to (89)–(90). The restriction to a finite index set of employed basis functions $\Lambda \subseteq \{(j,k) | j \geq j_0, k \in \mathcal{J}_j\}$ leads to the finite dimensional problem, required for the numerical treatment.

The above mentioned orthogonality and biorthogonality relations ensure that the Galerkin matrices obtained, when restricting (89)–(90) to such choices of subspaces Υ_Λ of Υ and $\mathcal{M}_\Lambda$ of $\mathcal{M}$, exhibit the sparseness structure illustrated by Figure 23. Of course, Λ is comprised here of finite index sets Λ_{x_i}, Λ_{y_i}, Λ_{v_i} and Λ_{w_i}. In general these index sets will result from an adaptive refinement of the trial spaces for each scalar unknown and may therefore develop, in principle, independently of each other. However, in order to simplify the technical exposition we will confine the discussion in the following to the case where all states x_i $(i = 1, \ldots n_x)$ have the same index set Λ_x, all outputs the index set Λ_y and likewise for the index sets of the remaining groups of components. Moreover, to ensure stability of the discretizations the spaces Υ_Λ, $\mathcal{M}_\Lambda$ have to satisfy the LBB-condition [35]. This means that $\mathcal{M}_\Lambda$ may not be too rich relative to Υ_Λ. Specifically, this requires that the index sets Λ'_x for the first group of Lagrange multipliers $\boldsymbol{\mu}$ acting on (85) should be contained in Λ_x. Likewise we will always make sure that $\Lambda_w \subset \Lambda_x$ and $\Lambda_v \subseteq \Lambda_y$. For a detailed discussion of an adaptive generation of the index sets we refer to [22].

Let us collect now in $\boldsymbol{d_x}$ all wavelet coefficients of the states $\boldsymbol{x}_\Lambda$ ($\boldsymbol{d_x} = \left(\boldsymbol{d_{x_i}}^\Lambda\right)_{i=1,\ldots n_x}$) and likewise for $\boldsymbol{d_y}, \boldsymbol{d_v}, \boldsymbol{d_w}$ where we have suppressed for convenience the index Λ. Inserting now the ansatz $\boldsymbol{\Upsilon}_\Lambda = (\boldsymbol{\Psi}^T_{\Lambda_x}\boldsymbol{d_x}, \boldsymbol{\Psi}^T_{\Lambda_y}\boldsymbol{d_y}, \boldsymbol{\Psi}^T_{\Lambda_v}\boldsymbol{d_v}, (\boldsymbol{\Psi}^H_{\Lambda_w})^T\boldsymbol{d_w}) \in \Upsilon_\Lambda$ and $\boldsymbol{\lambda}_\Lambda = ((\boldsymbol{\Psi}^H_{\Lambda'_x})^T\boldsymbol{d_\mu}, \tilde{\boldsymbol{\Psi}}^T_{\Lambda_y}\boldsymbol{d_\nu}) \in \mathcal{M}_\Lambda$ into (89)–(90) and testing with all the basis functions in Υ_Λ and $\mathcal{M}_\Lambda$ provides a linear system of equations, the KKT–system. Under the above conventions on the index sets the discretization matrix can be described in a rather condensed form with the aid of Kronecker products. First recall our shorthand notation for Gramian matrices e.g. $\langle\langle \boldsymbol{\Psi}_\Lambda, \boldsymbol{\Psi}_{\Lambda'}\rangle\rangle = \left(\langle \psi_{(j,k)}, \psi_{(j',k')}\rangle\right)_{(j,k)\in\Lambda,(j',k')\in\Lambda'}$. Moreover, the Kronecker product of a matrix $\boldsymbol{A} \in \mathbb{R}^{n\times m}$ with a matrix $\boldsymbol{B}$ is defined by $\boldsymbol{A} \otimes \boldsymbol{B} := (a_{i,j}\boldsymbol{B})_{\substack{i=1,\ldots,n\\ j=1,\ldots,m}}$. Defining now

$$\boldsymbol{Q}_\Lambda = \boldsymbol{Q} \otimes \langle\langle \boldsymbol{\Psi}_{\Lambda_y}, \boldsymbol{\Psi}_{\Lambda_y}\rangle\rangle, \quad \boldsymbol{R}_{\boldsymbol{v}_\Lambda} = \boldsymbol{R}_v \otimes \langle\langle \boldsymbol{\Psi}_{\Lambda_v}, \boldsymbol{\Psi}_{\Lambda_v}\rangle\rangle, \quad \boldsymbol{R}_{\boldsymbol{w}_\Lambda} = \boldsymbol{R}_w \otimes \boldsymbol{I}_{\Lambda_w}, \tag{93}$$

$$\boldsymbol{X}_\Lambda = \boldsymbol{I}_{n_x} \otimes \langle\langle \boldsymbol{\Psi}^H_{\Lambda'_x}, \dot{\boldsymbol{\Psi}}_{\Lambda_x}\rangle\rangle - \boldsymbol{A} \otimes \langle\langle \boldsymbol{\Psi}^H_{\Lambda'_x}, \boldsymbol{\Psi}_{\Lambda_x}\rangle\rangle, \ \boldsymbol{W}_\Lambda = \boldsymbol{W} \otimes \langle\langle \boldsymbol{\Psi}^H_{\Lambda'_x}, \boldsymbol{\Psi}^H_{\Lambda_w}\rangle\rangle, \tag{94}$$

$$\boldsymbol{I}_\Lambda = \boldsymbol{I}_{n_y} \otimes \boldsymbol{I}_{\Lambda_y}, \qquad \boldsymbol{C}_\Lambda = \boldsymbol{C} \otimes \langle\langle \tilde{\boldsymbol{\Psi}}_{\Lambda_y}, \boldsymbol{\Psi}_{\Lambda_x}\rangle\rangle, \quad \boldsymbol{V}_\Lambda = \boldsymbol{V} \otimes \langle\langle \tilde{\boldsymbol{\Psi}}_{\Lambda_y}, \boldsymbol{\Psi}_{\Lambda_v}\rangle\rangle \ , \tag{95}$$

$$\boldsymbol{B}_\Lambda = \boldsymbol{B} \otimes \langle\langle \boldsymbol{\Psi}^H_{\Lambda'_x}, \boldsymbol{\Psi}^H_{\Lambda_u}\rangle\rangle \tag{96}$$

and bearing in mind that on account of orthonormality and biorthogonality the wavelet mass matrices $\langle\langle \boldsymbol{\Psi}^H_{\Lambda'_x}, \boldsymbol{\Psi}^H_{\Lambda_w}\rangle\rangle$ and $\langle\langle \tilde{\boldsymbol{\Psi}}_{\Lambda'_y}, \boldsymbol{\Psi}_{\Lambda_v}\rangle\rangle$ are identity matrices possibly enlarged by a zero block, while the others exhibit finger structure, the restriction of (89)–(90) to $(\Upsilon_\Lambda, \mathcal{M}_\Lambda)$ yields the KKT–system:

$$\begin{pmatrix} & & & & {\boldsymbol{X}_\Lambda}^T & -{\boldsymbol{C}_\Lambda}^T \\ & 2\boldsymbol{Q}_\Lambda & & & & \boldsymbol{I}_\Lambda \\ & & 2\boldsymbol{R}_{\boldsymbol{v}_\Lambda} & & & -\boldsymbol{V}^T_\Lambda \\ & & & 2\boldsymbol{R}_{\boldsymbol{w}_\Lambda} & -{\boldsymbol{W}_\Lambda}^T & \\ \boldsymbol{X}_\Lambda & & & -\boldsymbol{W}_\Lambda & & \\ -\boldsymbol{C}_\Lambda & \boldsymbol{I}_\Lambda & -\boldsymbol{V}_\Lambda & & & \end{pmatrix} \begin{pmatrix} \boldsymbol{d_x} \\ \boldsymbol{d_y} \\ \boldsymbol{d_v} \\ \boldsymbol{d_w} \\ \boldsymbol{d_\mu} \\ \boldsymbol{d_\nu} \end{pmatrix} = \begin{pmatrix} \boldsymbol{0} \\ 2\boldsymbol{Q}_\Lambda\boldsymbol{d_z} \\ \boldsymbol{0} \\ \boldsymbol{0} \\ \boldsymbol{B}_\Lambda\boldsymbol{d_u} \\ \boldsymbol{0} \end{pmatrix}. \tag{97}$$

This linear system is simply the Galerkin discretization of (91) with respect to the finite dimensional spaces $\Upsilon_\Lambda, \mathcal{M}_\Lambda$. Of course, this system also arises from the necessary KKT condition for the discrete minimization problem

$$\min_{\boldsymbol{d_x},\,\boldsymbol{d_y},\,\boldsymbol{d_v},\,\boldsymbol{d_w}} \quad (\boldsymbol{d_y}-\boldsymbol{d_z})^T \boldsymbol{Q}_\Lambda (\boldsymbol{d_y}-\boldsymbol{d_z}) + \boldsymbol{d}_{\boldsymbol{v}}^T \boldsymbol{R}_{\boldsymbol{v}_\Lambda} \boldsymbol{d_v} + \boldsymbol{d}_{\boldsymbol{w}}^T \boldsymbol{R}_{\boldsymbol{w}_\Lambda} \boldsymbol{d_w}$$

subject to the constraints

$$\begin{aligned} \boldsymbol{X}_\Lambda \boldsymbol{d_x} - \boldsymbol{W}_\Lambda \boldsymbol{d_w} &= \boldsymbol{B}_\Lambda \boldsymbol{u}\,, \\ \boldsymbol{I}_\Lambda \boldsymbol{d_y} - \boldsymbol{C}_\Lambda \boldsymbol{d_x} - \boldsymbol{V}_\Lambda \boldsymbol{d_v} &= \boldsymbol{0}\,, \end{aligned}$$

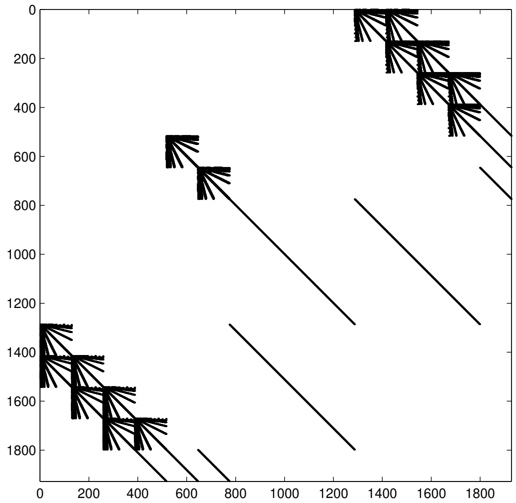
obtained by restricting the minimization in (84)–(86) to the finite dimensional space Υ_Λ.

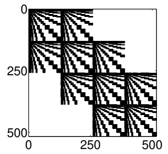
The linear system (97) has the following block structure

$$\boldsymbol{\mathcal{L}}_\Lambda \begin{pmatrix} \boldsymbol{\Upsilon}_\Lambda \\ \boldsymbol{\lambda}_\Lambda \end{pmatrix} := \begin{pmatrix} \boldsymbol{\mathcal{A}}_\Lambda & \boldsymbol{\mathcal{B}}_\Lambda{}^T \\ \boldsymbol{\mathcal{B}}_\Lambda & \boldsymbol{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{\Upsilon}_\Lambda \\ \boldsymbol{\lambda}_\Lambda \end{pmatrix} = \begin{pmatrix} \boldsymbol{h}_\Lambda \\ \mathbf{g}_\Lambda \end{pmatrix}, \tag{98}$$

where the constraints are represented by $\boldsymbol{\mathcal{B}}_\Lambda$ and the vector $\boldsymbol{\lambda}_\Lambda$ is comprised of the Lagrange parameters. However, here the matrix $\boldsymbol{\mathcal{A}}_\Lambda$ reflecting the objective function is only positive *semidefinite* since the states $\boldsymbol{x}$ are not regularized at this point. This is also illustrated by Figure 23a. It displays the nonzero pattern of the discretization matrix for a system with four states and one output function discretized on an equidistant grid of meshsize 2^{-7}.

Figure 23: 23a. KKT–matrix $\boldsymbol{\mathcal{L}}_\Lambda$ 23b. Schur complement $\boldsymbol{\mathcal{S}}_\Lambda$ (30).





4 Numerical Algorithm

In this section we want to outline the solution of the optimization problem (84) subject to the constraints (85)–(86), or equivalently, the corresponding operator equation (91). We employ a refinement approach to provide a state estimation at any instant of time. Therefore, we consider a sequence of finite dimensional linear systems (98)

$$\boldsymbol{\mathcal{L}}_{\Lambda^\ell} \begin{pmatrix} \boldsymbol{\Upsilon}_{\Lambda^\ell} \\ \boldsymbol{\lambda}_{\Lambda_\ell} \end{pmatrix} = \begin{pmatrix} \boldsymbol{h}_{\Lambda_\ell} \\ \mathbf{g}_{\Lambda^\ell} \end{pmatrix} \qquad \text{for } \ \ell = \ell_0, \ell_0 + 1, \ldots \tag{99}$$

and apply to the whole sequence of equations a *nested iteration* scheme as introduced for example in [35] (see Section 6 for a sketch). Moreover, each equation system is solved by an iterative solver, which we describe in the following subsection.

4.1 Simplified Uzawa Algorithm

To obtain an efficient iterative solver for a fixed level ℓ the structure of the system given by the original problem has to be exploited. The *Uzawa algorithm* is known as a classical scheme for saddle point problems of the form (98) [35, 38, 88]. However, a direct application of the Uzawa technique is prohibited by the rank deficiency of the upper left block $\boldsymbol{\mathcal{A}}_\Lambda$ in (98). Nevertheless, due to the simplicity of the nonzero part of $\boldsymbol{\mathcal{A}}_\Lambda$, we can easily reduce system (97) by eliminating the wavelet coefficients $\boldsymbol{d_y}, \boldsymbol{d_v}, \boldsymbol{d_w}$. These coefficients can be expressed by the dual solution $\boldsymbol{d_\mu}, \boldsymbol{d_\nu}$. The resulting equations are:

$$\begin{pmatrix} \frac{1}{2}\left(\boldsymbol{Q}_\Lambda^{-1} + \boldsymbol{V}_\Lambda \boldsymbol{R}_{\boldsymbol{v}_\Lambda}{}^{-1} \boldsymbol{V}_\Lambda{}^T\right) & \mathbf{0} & \boldsymbol{C}_\Lambda \\ \mathbf{0} & \frac{1}{2}\boldsymbol{W}_\Lambda \boldsymbol{R}_{\boldsymbol{w}_\Lambda}{}^{-1} \boldsymbol{W}_\Lambda{}^T & -\boldsymbol{X}_\Lambda \\ \boldsymbol{C}_\Lambda{}^T & -\boldsymbol{X}_\Lambda{}^T & \mathbf{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{d_\nu} \\ \boldsymbol{d_\mu} \\ \boldsymbol{d_x} \end{pmatrix} = \begin{pmatrix} \boldsymbol{d_z} \\ -\boldsymbol{B}_\Lambda \boldsymbol{d_u} \\ \mathbf{0} \end{pmatrix} \tag{100}$$

and

$$\begin{aligned} 2\boldsymbol{Q}_\Lambda \boldsymbol{d_y} &= -\boldsymbol{d_\nu} + 2\boldsymbol{Q}_\Lambda \boldsymbol{d_z} \ , && (101) \\ 2\boldsymbol{R}_{\boldsymbol{v}_\Lambda} \boldsymbol{d_v} &= \boldsymbol{V}_\Lambda{}^T \boldsymbol{d_\nu} \ , && (102) \\ 2\boldsymbol{R}_{\boldsymbol{w}_\Lambda} \boldsymbol{d_w} &= \boldsymbol{W}_\Lambda{}^T \boldsymbol{d_\mu} \ . && (103) \end{aligned}$$

The equations (101)–(103) can be easily solved by exploiting the rule $(\boldsymbol{E} \otimes \boldsymbol{F})^{-1} = \boldsymbol{E}^{-1} \otimes \boldsymbol{F}^{-1}$ for the inversion of $\boldsymbol{Q}_\Lambda$, $\boldsymbol{R}_{\boldsymbol{v}_\Lambda}$ and $\boldsymbol{R}_{\boldsymbol{w}_\Lambda}$. Moreover, recall from (93)–(96) that the weight matrices $\boldsymbol{Q}$, $\boldsymbol{R}_v$ and $\boldsymbol{R}_w$ are in general diagonal. Furthermore, due to the stability of the wavelet bases the matrices $\langle\langle \boldsymbol{\Psi}_{\Lambda_y}, \boldsymbol{\Psi}_{\Lambda_y} \rangle\rangle$ and $\langle\langle \boldsymbol{\Psi}_{\Lambda_v}, \boldsymbol{\Psi}_{\Lambda_v} \rangle\rangle$ are well–conditioned so that the systems involving these latter matrices can be solved very efficiently by an iterative scheme.

System (100) has again the structure (98)

$$\tilde{\boldsymbol{\mathcal{L}}}_\Lambda \begin{pmatrix} \tilde{\boldsymbol{\lambda}}_\Lambda \\ \tilde{\boldsymbol{\Upsilon}}_\Lambda \end{pmatrix} = \begin{pmatrix} \tilde{\boldsymbol{\mathcal{A}}}_\Lambda & \tilde{\boldsymbol{\mathcal{B}}}_\Lambda{}^T \\ \tilde{\boldsymbol{\mathcal{B}}}_\Lambda & \mathbf{0} \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{\lambda}}_\Lambda \\ \tilde{\boldsymbol{\Upsilon}}_\Lambda \end{pmatrix} = \begin{pmatrix} \tilde{\boldsymbol{h}}_\Lambda \\ \tilde{\mathbf{g}}_\Lambda \end{pmatrix} , \tag{104}$$

where $\tilde{\boldsymbol{\lambda}}_\Lambda = (\boldsymbol{d}_{\boldsymbol{\nu}}^T, \boldsymbol{d}_{\boldsymbol{\mu}}^T)^T$ and $\tilde{\boldsymbol{\Upsilon}}_\Lambda = \boldsymbol{d}_{\boldsymbol{x}}$. The corresponding infinite dimensional system is denoted by

$$\tilde{\boldsymbol{\mathcal{L}}} \begin{pmatrix} \tilde{\boldsymbol{\lambda}} \\ \tilde{\boldsymbol{\Upsilon}} \end{pmatrix} = \begin{pmatrix} \tilde{\boldsymbol{h}} \\ \tilde{\mathrm{g}} \end{pmatrix}. \tag{105}$$

Considering $\tilde{\boldsymbol{\mathcal{A}}}_\Lambda$ in more detail, we observe that $\left(\boldsymbol{Q}_\Lambda^{-1} + \boldsymbol{V}_\Lambda \boldsymbol{R}_{\boldsymbol{v}_\Lambda}{}^{-1} \boldsymbol{V}_\Lambda{}^T\right)$ has full rank and is positive definite. So is $\boldsymbol{W}_\Lambda \boldsymbol{R}_{\boldsymbol{w}_\Lambda}{}^{-1} \boldsymbol{W}_\Lambda{}^T$, provided that $\boldsymbol{W}$ has the same rank as $\boldsymbol{A}$ ($n_w = n_x$). However, since the KKT–matrix in (97) has always full rank for a correctly posed problem, the following ideas can be extended to the more general case.

Now $\tilde{\boldsymbol{\mathcal{A}}}_\Lambda$ is positive definite so that the following Uzawa technique can be applied. Recall that this algorithm is based on the reformulation of (104) to

$$\begin{aligned} \tilde{\boldsymbol{\mathcal{A}}}_\Lambda \tilde{\boldsymbol{\lambda}}_\Lambda &= \tilde{\boldsymbol{h}}_\Lambda - \tilde{\boldsymbol{\mathcal{B}}}_\Lambda{}^T \tilde{\boldsymbol{\Upsilon}}_\Lambda & (106) \\ \tilde{\boldsymbol{\mathcal{B}}}_\Lambda \tilde{\boldsymbol{\mathcal{A}}}_\Lambda{}^{-1} \tilde{\boldsymbol{\mathcal{B}}}_\Lambda{}^T \tilde{\boldsymbol{\Upsilon}}_\Lambda &= \tilde{\boldsymbol{\mathcal{B}}}_\Lambda \tilde{\boldsymbol{\mathcal{A}}}_\Lambda{}^{-1} \tilde{\boldsymbol{h}}_\Lambda - \tilde{\mathrm{g}}_\Lambda & (107) \end{aligned}$$

by block elimination. The system (107) involving the Schur complement $\tilde{\boldsymbol{\mathcal{B}}}_\Lambda \tilde{\boldsymbol{\mathcal{A}}}_\Lambda{}^{-1} \tilde{\boldsymbol{\mathcal{B}}}_\Lambda{}^T$ is solved by the pcg–method. However, since generally the Schur complement involves the inverse of the upper left block $\tilde{\boldsymbol{\mathcal{A}}}_\Lambda$ it is usually not computed explicitly. Instead each pcg–step for (107) requires the solution of a linear system involving $\tilde{\boldsymbol{\mathcal{A}}}_\Lambda$, which is done again by applying the pcg–method as an inner iteration. Moreover, this algorithm for (107) yields also in each step the search direction for solving (106). In this sense the Uzawa algorithm combines the solution of (107) and (106), avoiding an additional application of the pcg–method to (106).

In our particular case this procedure can be simplified since the inverse $\tilde{\boldsymbol{\mathcal{A}}}_\Lambda{}^{-1}$ can be computed explicitly by exploiting the Kronecker product. In fact, one has

$$\tilde{\boldsymbol{\mathcal{A}}}_\Lambda{}^{-1} = \begin{pmatrix} \left(\boldsymbol{Q}^{-1} + \boldsymbol{V}\boldsymbol{R}_v{}^{-1}\boldsymbol{V}^T\right)^{-1} \otimes \langle\langle \boldsymbol{\Psi}_{\Lambda_y}, \boldsymbol{\Psi}_{\Lambda_y} \rangle\rangle & \boldsymbol{0} \\ \boldsymbol{0} & (\boldsymbol{W}\boldsymbol{R}_w{}^{-1}\boldsymbol{W}^T)^{-1} \otimes \boldsymbol{I}_{\Lambda_x'} \end{pmatrix}. \tag{108}$$

Note that the inverses $\left(\boldsymbol{Q}^{-1} + \boldsymbol{V}\boldsymbol{R}_v{}^{-1}\boldsymbol{V}^T\right)^{-1}$ and $(\boldsymbol{W}\boldsymbol{R}_w{}^{-1}\boldsymbol{W}^T)^{-1}$ can be computed once and for all in advance since they are independent of the current discretization.

This leads to the following *simplified Uzawa algorithm*. As in the classical version the pcg–method is applied to the Schur complement $\boldsymbol{\mathcal{S}}_\Lambda = \tilde{\boldsymbol{\mathcal{B}}}_\Lambda \tilde{\boldsymbol{\mathcal{A}}}_\Lambda{}^{-1} \tilde{\boldsymbol{\mathcal{B}}}_\Lambda^T$. Of course, the matrix $\boldsymbol{\mathcal{S}}_\Lambda$ is not assembled. Instead only matrix–vector multiplications involving the matrices $\tilde{\boldsymbol{\mathcal{A}}}_\Lambda{}^{-1}$, $\tilde{\boldsymbol{\mathcal{B}}}_\Lambda$ and $\tilde{\boldsymbol{\mathcal{B}}}_\Lambda^T$ are performed as detailed below in (109). The inner iterations of the Uzawa algorithm, namely the application of the pcg–method to the systems involving $\tilde{\boldsymbol{\mathcal{A}}}_\Lambda$, are substituted by a matrix multiplication with $\tilde{\boldsymbol{\mathcal{A}}}_\Lambda{}^{-1}$. Therefore preconditioning refers in this algorithm only to the Schur complement. Like for the pcg–method it requires an appropriate positive definite matrix $\boldsymbol{M}$ applied to the current residuals as shown in (109) below. Thus we obtain the following algorithm for the system (104). To avoid a severe cluttering of indices we suppress here the index Λ and the superscript $\tilde{}$.

Initialization: Let $\boldsymbol{\Upsilon}_0$ be given, then

$$\begin{aligned}
\boldsymbol{\lambda}_0 &= \boldsymbol{\mathcal{A}}^{-1}\left(\boldsymbol{h} - \boldsymbol{\mathcal{B}}^T\boldsymbol{\Upsilon}_0\right) \\
\boldsymbol{r}_0 &= \mathrm{g} - \boldsymbol{\mathcal{B}}\boldsymbol{\lambda}_0 \\
\boldsymbol{q}_0 &= \boldsymbol{M}\boldsymbol{r}_0 \\
\boldsymbol{d}_0 &= -\boldsymbol{q}_0 \\
\rho_0 &= \boldsymbol{r}_0^T\boldsymbol{q}_0
\end{aligned}$$

Iterations: For $k = 0, 1, \ldots$ compute:

$$\begin{aligned}
\boldsymbol{p}_k &= \boldsymbol{\mathcal{B}}^T\boldsymbol{d}_k \\
\boldsymbol{s}_k &= \boldsymbol{\mathcal{A}}^{-1}\boldsymbol{p}_k \\
\alpha_k &= \rho_k/(\boldsymbol{p}_k^T\boldsymbol{s}_k) \\
\boldsymbol{\Upsilon}_{k+1} &= \boldsymbol{\Upsilon}_k + \alpha_k\boldsymbol{d}_k \\
\boldsymbol{\lambda}_{k+1} &= \boldsymbol{\lambda}_k - \alpha_k\boldsymbol{s}_k \\
\boldsymbol{r}_{k+1} &= \mathrm{g} - \boldsymbol{\mathcal{B}}\boldsymbol{\lambda}_{k+1} \\
\boldsymbol{q}_{k+1} &= \boldsymbol{M}\boldsymbol{r}_{k+1} \\
\rho_{k+1} &= \boldsymbol{r}_{k+1}^T\boldsymbol{q}_{k+1} \\
\boldsymbol{d}_{k+1} &= -\boldsymbol{q}_{k+1} + \rho_{k+1}/\rho_k\ \boldsymbol{d}_k\,.
\end{aligned} \tag{109}$$

Note that this algorithm determines not only $\tilde{\boldsymbol{\Upsilon}}_\Lambda = \boldsymbol{d_x}$ but also the unknown $\tilde{\boldsymbol{\lambda}}_\Lambda = (\boldsymbol{d}_{\boldsymbol{\nu}}^T, \boldsymbol{d}_{\boldsymbol{\mu}}^T)^T$ iteratively.

To obtain a rough estimate for the computational cost of this scheme let $nz(\boldsymbol{A})$ denote the number of nonzero entries in a matrix $\boldsymbol{A}$. Then for k iterations the cost of the algorithm is $k\ (\ 2rank\ \boldsymbol{\mathcal{B}} - 2 + nz(\boldsymbol{\mathcal{A}}^{-1}) + 2nz(\boldsymbol{\mathcal{B}}) + nz(\boldsymbol{M})\ ) - 2rank\ \boldsymbol{\mathcal{B}} - rank\ \boldsymbol{\mathcal{A}} + 1$ additions and $k\ (\ 3rank\ \boldsymbol{\mathcal{B}} + 2rank\ \boldsymbol{\mathcal{A}} + 2 + nz(\boldsymbol{\mathcal{A}}^{-1}) + 2nz(\boldsymbol{\mathcal{B}}) + nz(\boldsymbol{M})\ ) - 2rank\ \boldsymbol{\mathcal{B}} - 2rank\ \boldsymbol{\mathcal{A}} - 2$ multiplications. For our application we have $rank\ \boldsymbol{\mathcal{B}} = n_x\#\Lambda_x$ and $rank\ \boldsymbol{\mathcal{A}} = n_y\#\Lambda_y + n_x\#\Lambda'_x$. Furthermore, taking into account that $\boldsymbol{\mathcal{B}}$ consists of blocks with finger structure originating from the wavelet discretization, $nz(\boldsymbol{\mathcal{B}})$ is of order $n_S\ O(\#\Lambda_x\log(\#\Lambda_x))$ (see [56]), where n_S is at most $nz(\boldsymbol{A}) + nz(\boldsymbol{C}) + n_x$. In general n_S is of order n_x due to the sparsity of the model matrices, but in the worst case it can be n_x^2. Assuming that the preconditioning matrix $\boldsymbol{M}$ is sparse, more specifically has at most $c\ rank\ \boldsymbol{\mathcal{B}}$ nonzero entries, the method requires $n_S\ O(\#\Lambda_x\log(\#\Lambda_x))$ flops per iteration.

Of course, one hopes to find a sufficiently good preconditioner $\boldsymbol{M}$ to ensure that the number k of iterations needed to achieve a desired accuracy stays far below the number of unknowns which is here $rank\ \boldsymbol{\mathcal{B}} = n_x\#\Lambda_x$. In fact, one has the following classical estimate for the iterates $\boldsymbol{d}_{\boldsymbol{x}}^{(k)} = \tilde{\boldsymbol{\Upsilon}}_\Lambda^{(k)}$ in terms of the spectral condition numbers $\kappa_2 := \mathrm{cond}_2(\boldsymbol{M}^{1/2}\boldsymbol{\mathcal{S}}_\Lambda\boldsymbol{M}^{1/2})$:

$$\|\ \boldsymbol{d}_{\boldsymbol{x}}^{(k)} - \boldsymbol{d}_{\boldsymbol{x}}\ \|_{\boldsymbol{\mathcal{S}}_\Lambda} \leq 2\ \|\ \boldsymbol{d}_{\boldsymbol{x}}^{(0)} - \boldsymbol{d}_{\boldsymbol{x}}\ \|_{\boldsymbol{\mathcal{S}}_\Lambda}\ \delta^k \tag{110}$$

where $\delta = \frac{\sqrt{\kappa_2}-1}{\sqrt{\kappa_2}+1}$ and where the *energy–norm* $\| \boldsymbol{d_x} \|_{\boldsymbol{A}}^2$ is defined as $\boldsymbol{d_x}^T \boldsymbol{A} \boldsymbol{d_x}$ [89].

As a consequence of the norm equivalence for wavelets [56]

$$c_1 \| \boldsymbol{d}_f \|_{\ell_2} \leq \| f \|_{L_2} \leq c_2 \| \boldsymbol{d}_f \|_{\ell_2} \tag{111}$$

the above estimates immediately yield the following error bound for the iterates $\boldsymbol{x}_\Lambda^{(k)} = \boldsymbol{\Psi}_\Lambda^T \boldsymbol{d}_{\boldsymbol{x}}^{(k)}$ as approximations to the function $\boldsymbol{x}_\Lambda = \boldsymbol{\Psi}_\Lambda^T \boldsymbol{d_x}$ in the energy–norm $\| \boldsymbol{x} \|_{\bar{\boldsymbol{S}}_\Lambda}^2 := \langle \boldsymbol{x}, \bar{\boldsymbol{S}}_\Lambda \boldsymbol{x} \rangle = \| \bar{\boldsymbol{S}}_\Lambda^{1/2} \boldsymbol{x} \|_{L_2^{n_x}}^2$ which is induced by the corresponding Schur complement $\bar{\boldsymbol{S}}_\Lambda$:

$$\| \boldsymbol{x}_\Lambda^{(k)} - \boldsymbol{x}_\Lambda \|_{\bar{\boldsymbol{S}}_\Lambda} \leq 2\bar{c} \| \boldsymbol{x}_\Lambda^{(0)} - \boldsymbol{x}_\Lambda \|_{\bar{\boldsymbol{S}}_\Lambda} \delta^k. \tag{112}$$

The constant $\bar{c} = \mathrm{cond}(\langle\langle \boldsymbol{\Psi}, \boldsymbol{\Psi} \rangle\rangle)$ is less than 12 for our choice of wavelets. The design of preconditioners that give rise to possibly small reduction factors δ, and hence to small iteration numbers, will be explained later, see Section 5.

4.2 Schur Complement

To prepare the ground for discussing the issue of preconditioning we consider the Schur complement in more detail. The reduction of system (97) to the unknowns $\boldsymbol{d_x}$ gives the Schur complement

$$\boldsymbol{S}_\Lambda = 2 \left[\boldsymbol{X}_\Lambda^T \left(\boldsymbol{W}_\Lambda \boldsymbol{R}_{\boldsymbol{w}_\Lambda}{}^{-1} \boldsymbol{W}_\Lambda^T \right)^{-1} \boldsymbol{X}_\Lambda + \boldsymbol{C}_\Lambda^T \left(\boldsymbol{Q}_\Lambda^{-1} + \boldsymbol{V}_\Lambda \boldsymbol{R}_{\boldsymbol{v}_\Lambda}{}^{-1} \boldsymbol{V}_\Lambda^T \right)^{-1} \boldsymbol{C}_\Lambda \right] \tag{113}$$

which is positive definite. The corresponding linear system is

$$1/2\, \boldsymbol{S}_\Lambda \boldsymbol{d_x} = \boldsymbol{X}_\Lambda^T \left(\boldsymbol{W}_\Lambda \boldsymbol{R}_{\boldsymbol{w}_\Lambda}{}^{-1} \boldsymbol{W}_\Lambda^T \right)^{-1} \boldsymbol{B}_\Lambda \boldsymbol{d_u} + \boldsymbol{C}_\Lambda^T \left(\boldsymbol{Q}_\Lambda^{-1} + \boldsymbol{V}_\Lambda \boldsymbol{R}_{\boldsymbol{v}_\Lambda}{}^{-1} \boldsymbol{V}_\Lambda^T \right)^{-1} \boldsymbol{d_z} \tag{114}$$

Abbreviating $\bar{\boldsymbol{Q}} := (\boldsymbol{Q}^{-1} + \boldsymbol{V} \boldsymbol{R}_v{}^{-1} \boldsymbol{V}^T)^{-1}$ and $\bar{\boldsymbol{R}}_{\boldsymbol{w}} := (\boldsymbol{W} \boldsymbol{R}_w{}^{-1} \boldsymbol{W}^T)^{-1}$ and exploiting the rules for the Kronecker product we derive from (93)–(96) and (108) the following representation of $\boldsymbol{S}_\Lambda$

$$\begin{aligned}
1/2\, \boldsymbol{S}_\Lambda \;=\; & \bar{\boldsymbol{R}}_{\boldsymbol{w}} \otimes \left(\langle\langle \dot{\boldsymbol{\Psi}}_{\Lambda_x}, \boldsymbol{\Psi}_{\Lambda_x'}^H \rangle\rangle \langle\langle \boldsymbol{\Psi}_{\Lambda_x'}^H, \dot{\boldsymbol{\Psi}}_{\Lambda_x} \rangle\rangle \right) \\
& - \left\{ \left(\boldsymbol{A}^T \bar{\boldsymbol{R}}_{\boldsymbol{w}} \right) \otimes \left(\langle\langle \boldsymbol{\Psi}_{\Lambda_x}, \boldsymbol{\Psi}_{\Lambda_x'}^H \rangle\rangle \langle\langle \boldsymbol{\Psi}_{\Lambda_x'}^H, \dot{\boldsymbol{\Psi}}_{\Lambda_x} \rangle\rangle \right) \right\} \\
& - \left\{ \left(\boldsymbol{A}^T \bar{\boldsymbol{R}}_{\boldsymbol{w}} \right) \otimes \left(\langle\langle \boldsymbol{\Psi}_{\Lambda_x}, \boldsymbol{\Psi}_{\Lambda_x'}^H \rangle\rangle \langle\langle \boldsymbol{\Psi}_{\Lambda_x'}^H, \dot{\boldsymbol{\Psi}}_{\Lambda_x} \rangle\rangle \right) \right\}^T \\
& + \left(\boldsymbol{A}^T \bar{\boldsymbol{R}}_{\boldsymbol{w}} \boldsymbol{A} \right) \otimes \left(\langle\langle \boldsymbol{\Psi}_{\Lambda_x}, \boldsymbol{\Psi}_{\Lambda_x'}^H \rangle\rangle \langle\langle \boldsymbol{\Psi}_{\Lambda_x'}^H, \boldsymbol{\Psi}_{\Lambda_x} \rangle\rangle \right) \\
& + \left(\boldsymbol{C}^T \bar{\boldsymbol{Q}} \boldsymbol{C} \right) \otimes \left(\langle\langle \boldsymbol{\Psi}_{\Lambda_x}, \tilde{\boldsymbol{\Psi}}_{\Lambda_y} \rangle\rangle \langle\langle \boldsymbol{\Psi}_{\Lambda_y}, \boldsymbol{\Psi}_{\Lambda_y} \rangle\rangle \langle\langle \tilde{\boldsymbol{\Psi}}_{\Lambda_y}, \boldsymbol{\Psi}_{\Lambda_x} \rangle\rangle \right) .
\end{aligned} \tag{115}$$

Note that all the wavelet matrices appearing in (115) are singular except those corresponding to $\boldsymbol{C}^T \bar{\boldsymbol{Q}} \boldsymbol{C}$. In fact, the necessary variation of initial values requires reserving at least one degree of freedom in each state variable $\boldsymbol{x}_i$ in excess of the test conditions. Moreover, in general, the last summand is also singular due to the singularity of $\boldsymbol{C}$. Figure 23b. exhibits the nonzero entries of $\boldsymbol{S}_\Lambda$ corresponding to the KKT–system in Figure 23a. to illustrate a typical situation.

5 Preconditioning

A successful application of the simplified Uzawa scheme hinges on finding an efficient preconditioner. Thus, it is important to understand two characteristic features of the mapping $\boldsymbol{\mathcal{L}}$ in (91) and its finite dimensional approximation $\boldsymbol{\mathcal{L}}_\Lambda$ in (98). Let $\sigma_{\max}(\Lambda)$, $\sigma_{\min}(\Lambda)$ denote the largest respectively smallest singular value of $\boldsymbol{\mathcal{L}}_\Lambda$. One usually faces the following facts:

(I) $\lim \sigma_{\max}(\Lambda) \to \infty$ as $\#\Lambda \to \infty$.

(II) It may happen that $\sigma_{\min}(\Lambda)$ is very small for all Λ.

Hence, system (97) and its modifications are generally very ill–conditioned. Property (I) is simply a consequence of the fact that the constraints involve derivatives of the states. Thus, although according to (92) $\boldsymbol{\mathcal{L}}$ is boundedly invertible as a mapping from $\Upsilon \times \mathcal{M}$ onto its dual it does not give rise to a well posed operator equation on L_2. On the other hand, its discrete approximation $\boldsymbol{\mathcal{L}}_\Lambda$ acts on wavelet coefficients which, by the Riesz basis property relate the Euclidean norm directly to the L_2–norm. Hence the spectral norms of the operators $\boldsymbol{\mathcal{L}}_\Lambda$ will grow with increasing dimension of the trial spaces. However, as we will see in the following, based upon norm equivalences for Sobolev spaces with properly scaled wavelet coefficients (see (116), (117)) one can choose respectively scaled wavelets, leading to a discretization of $\boldsymbol{\mathcal{L}}$ with bounded spectral norm. This scaling can also be interpreted as symmetric preconditioning of $\boldsymbol{\mathcal{L}}_\Lambda$, as we do in the context of iterative solvers.

The reason for property (II) is completely different. The original process model has often a low observability measure. Large changes in $\boldsymbol{x}$ may correspond to only very small changes in $\boldsymbol{y}$ and, respectively, in $\boldsymbol{z}$. This requires additional efforts concerning preconditioning and its combination with a suitable choice of regularization parameters $\boldsymbol{Q}, \boldsymbol{R}_v$ and $\boldsymbol{R}_w$. First approaches are given in Part 2 [27].

To this end, recall from [59] that the piecewise linear wavelet bases used in Section 3 for the discretization of the states induces the following isomorphism between ℓ_2 and H^1. There exist positive bounded constants c_3, c_4 such that for any $f \in H^1$ whose wavelet expansion is $f = \sum_{j,k} d_{j,k}\psi_{j,k}$ (i.e., $d_{j,k} = \langle f, \tilde{\psi}_{j,k}\rangle$), one has

$$c_3 \|\{2^j d_{j,k}\}_{j,k}\|_{\ell_2} \leq \left(\|f\|_{L_2}^2 + \|\dot{f}\|_{L_2}^2\right)^{1/2} = \|f\|_{H^1} \leq c_4 \|\{2^j d_{j,k}\}_{j,k}\|_{\ell_2}. \tag{116}$$

By duality for elements in the dual space $f \in (H^1)'$ and their dual wavelet expansions $f = \sum_{j,k} \tilde{d}_{j,k}\tilde{\psi}_{j,k}$ the following norm equivalence holds

$$c_5 \|\{2^{-j}\tilde{d}_{j,k}\}_{j,k}\|_{\ell_2} \leq \|f\|_{(H^1)'} \leq c_6 \|\{2^{-j}\tilde{d}_{j,k}\}_{j,k}\|_{\ell_2}. \tag{117}$$

Now let $\boldsymbol{D}_\Lambda$ denote the diagonal matrix with diagonal entries 2^j for $(j,k) \in \Lambda_x$. We will indicate next that these norm equivalences imply bounds on the condition numbers of properly scaled Schur complements (see [56, 57] for related discussions).

Theorem 5.1 If the operator $\boldsymbol{\mathcal{L}}$ is well–posed with respect to $\Upsilon \times \mathcal{M}$ and provided the used Galerkin scheme is stable, then

$$\mathrm{cond}_2\left(\boldsymbol{D}_\Lambda^{-1}\boldsymbol{\mathcal{S}}_\Lambda\boldsymbol{D}_\Lambda^{-1}\right) = \mathcal{O}(1) \tag{118}$$

independently of Λ.

Proof: The operator $\tilde{\boldsymbol{\mathcal{L}}}$ (see (105)) maps $(L_2^{n_y} \times L_2^{n_w}) \times (H^1)^{n_x}$ into its dual $(L_2^{n_y} \times L_2^{n_w}) \times ((H^1)^{n_x})'$. Given the fact that $\boldsymbol{\mathcal{L}}$ is well–posed with respect to $\Upsilon \times \mathcal{M}$, one concludes from (92) that there exist positive constants c_7, c_8 for $\tilde{\boldsymbol{\mathcal{L}}}$ such that

$$\begin{aligned} c_7\left(\|\tilde{\boldsymbol{\lambda}}\|^2_{(L_2^{n_y}\times L_2^{n_w})} + \|\boldsymbol{x}\|^2_{(H^1)^{n_x})}\right)^{1/2} &\leq \|\tilde{\mathcal{L}}\begin{pmatrix}\tilde{\boldsymbol{\lambda}}\\ \boldsymbol{x}\end{pmatrix}\|_{(L_2^{n_y}\times L_2^{n_w})\times((H^1)^{n_x})'} \\ &\leq c_8\left(\|\tilde{\boldsymbol{\lambda}}\|^2_{(L_2^{n_y}\times L_2^{n_w})} + \|\boldsymbol{x}\|^2_{(H^1)^{n_x})}\right)^{1/2}. \end{aligned} \tag{119}$$

Now expand the functions $\tilde{\boldsymbol{\lambda}}, \boldsymbol{x}$ in the respective *infinite* dimensional wavelet bases, namely $\tilde{\boldsymbol{\lambda}} = (\boldsymbol{d}_{\boldsymbol{\nu}}^T\tilde{\boldsymbol{\Psi}}, \boldsymbol{d}_{\boldsymbol{\mu}}^T\boldsymbol{\Psi}^H)$ and $\boldsymbol{x} = \hat{\boldsymbol{d}}_{\boldsymbol{x}}^{\,T}\boldsymbol{D}^{-1}\boldsymbol{\Psi}$. Note that we have scaled the basis for the $\boldsymbol{x}$–component so that in our earlier notation $\hat{\boldsymbol{d}}_{\boldsymbol{x}} = \boldsymbol{D}\boldsymbol{d}_{\boldsymbol{x}}$. Invoking the Riesz basis property (111) for the first two components and employing (116) for the third component, the latter scaling of $\boldsymbol{\Psi}$ just ensures that

$$\begin{aligned} c_3\left(\|\boldsymbol{d}_{\boldsymbol{\nu}}\|^2_{\ell_2} + \|\boldsymbol{d}_{\boldsymbol{\mu}}\|^2_{\ell_2} + \|\hat{\boldsymbol{d}}_{\boldsymbol{x}}\|^2_{\ell_2}\right)^{1/2} &\leq \left(\|\tilde{\boldsymbol{\lambda}}\|^2_{L_2} + \|\boldsymbol{x}\|^2_{H^1}\right)^{1/2} \\ &\leq c_4\left(\|\boldsymbol{d}_{\boldsymbol{\nu}}\|^2_{\ell_2} + \|\boldsymbol{d}_{\boldsymbol{\mu}}\|^2_{\ell_2} + \|\hat{\boldsymbol{d}}_{\boldsymbol{x}}\|^2_{\ell_2}\right)^{1/2}. \end{aligned} \tag{120}$$

Moreover, since the norm equivalences (111) and (117) give rise to

$$\begin{aligned} 1/c_9\left\|\tilde{\mathcal{L}}\begin{pmatrix}\tilde{\boldsymbol{\lambda}}\\ \boldsymbol{x}\end{pmatrix}\right\|_{L_2^{n_y}\times L_2^{n_w}\times((H^1)^{n_x})'} &\leq \left\|\begin{pmatrix}\tilde{\boldsymbol{\mathcal{A}}} & \tilde{\boldsymbol{\mathcal{B}}}^T\boldsymbol{D}^{-1}\\ \boldsymbol{D}^{-1}\tilde{\boldsymbol{\mathcal{B}}} & \boldsymbol{0}\end{pmatrix}\begin{pmatrix}(\boldsymbol{d}_{\boldsymbol{\nu}}^T, \boldsymbol{d}_{\boldsymbol{\mu}}^T)^T\\ \hat{\boldsymbol{d}}_{\boldsymbol{x}}\end{pmatrix}\right\|_{\ell_2} \\ &\leq 1/c_{10}\left\|\tilde{\mathcal{L}}\begin{pmatrix}\tilde{\boldsymbol{\lambda}}\\ \boldsymbol{x}\end{pmatrix}\right\|_{L_2^{n_y}\times L_2^{n_w}\times((H^1)^{n_x})'} \end{aligned} \tag{121}$$

with $c_9 = \max\{c_2, c_6\}$ and $c_{10} = \min\{c_1, c_5\}$, we conclude from (119), (120) and (121) that

$$\begin{aligned} c_3c_7/c_9\left(\|\boldsymbol{d}_{\boldsymbol{\nu}}\|^2_{\ell_2} + \|\boldsymbol{d}_{\boldsymbol{\mu}}\|^2_{\ell_2} + \|\hat{\boldsymbol{d}}_{\boldsymbol{x}}\|^2_{\ell_2}\right)^{1/2} &\leq \left\|\begin{pmatrix}\tilde{\boldsymbol{\mathcal{A}}} & \tilde{\boldsymbol{\mathcal{B}}}^T\boldsymbol{D}^{-1}\\ \boldsymbol{D}^{-1}\tilde{\boldsymbol{\mathcal{B}}} & \boldsymbol{0}\end{pmatrix}\begin{pmatrix}(\boldsymbol{d}_{\boldsymbol{\nu}}^T, \boldsymbol{d}_{\boldsymbol{\mu}}^T)^T\\ \hat{\boldsymbol{d}}_{\boldsymbol{x}}\end{pmatrix}\right\|_{\ell_2} \\ &\leq c_4c_8/c_{10}\left(\|\boldsymbol{d}_{\boldsymbol{\nu}}\|^2_{\ell_2} + \|\boldsymbol{d}_{\boldsymbol{\mu}}\|^2_{\ell_2} + \|\hat{\boldsymbol{d}}_{\boldsymbol{x}}\|^2_{\ell_2}\right)^{1/2}. \end{aligned} \tag{122}$$

This in turn means that the *infinite* matrix $\begin{pmatrix}\tilde{\boldsymbol{\mathcal{A}}} & \tilde{\boldsymbol{\mathcal{B}}}^T\boldsymbol{D}^{-1}\\ \boldsymbol{D}^{-1}\tilde{\boldsymbol{\mathcal{B}}} & \boldsymbol{0}\end{pmatrix}$ as well as its inverse are bounded in Euclidean metric.

Next recall that the relation (122) remains valid for truncated finite sequences $\boldsymbol{d}_{\boldsymbol{\nu}}, \boldsymbol{d}_{\boldsymbol{\mu}}, \hat{\boldsymbol{d}}_{\boldsymbol{x}}$ determined by some finite index set Λ and for the corresponding finite matrices $\begin{pmatrix} \tilde{\boldsymbol{\mathcal{A}}}_\Lambda & \tilde{\boldsymbol{\mathcal{B}}}_\Lambda{}^T \boldsymbol{D}_\Lambda^{-1} \\ \boldsymbol{D}_\Lambda^{-1}\tilde{\boldsymbol{\mathcal{B}}}_\Lambda & \boldsymbol{0} \end{pmatrix}$ *uniformly* in Λ provided that the Galerkin scheme associated with these trial spaces is stable. Hence the matrices remain uniformly bounded. Consequently, we can deduce $\boldsymbol{D}_\Lambda^{-1}\boldsymbol{\mathcal{S}}_\Lambda \boldsymbol{D}_\Lambda^{-1}$ is uniformly bounded, which in turn proofs the theorem. q.e.d.

In the present context the Galerkin scheme is stable if and only if the trial spaces for the first group of variables $\tilde{\boldsymbol{\lambda}}$ on one hand and for the second group of variables $\boldsymbol{x}$ on the other hand, satisfy the so called LBB–condition, see [35, 38]. The validity of the LBB–condition can be ensured by keeping the refinement level for the first group of variables, i.e. the index sets for the Lagrange multipliers, in relation to the second group, e.g. as chosen.

The quantitative behavior of the above scaling $\boldsymbol{D}^{-1}$ of the wavelet coefficients and, consequently, of the corresponding preconditioner $\boldsymbol{M} = \boldsymbol{D}^{-2}$ for the Uzawa algorithm can still be improved. Note that (116) means that a properly scaled wavelet basis is a Riesz basis for the Sobolev space H^1. Therefore one expects that the Riesz constants $c_1, \ldots, c_6$ (and hence the effect on preconditioning) in (111), (116) and (117) can be improved by normalizing the wavelet basis in H^1 or better yet in the relevant energy space defined by the Schur complement. This suggests to employ as a diagonal preconditioner the matrix $\boldsymbol{M}_\Lambda^J$ obtained by taking the inverses of the diagonal entries of the Schur complement. Thus, wavelet theory proves that the application of the Jacobi preconditioner $\boldsymbol{M}_\Lambda^J$ is sufficient for bounded conditioned numbers. Recall that we wish to avoid expensive matrix multiplications. Therefore, we consider the option of *approximating* the diagonal entries of the Schur complement while avoiding its explicit computation. The involved inverse $\tilde{\boldsymbol{\mathcal{A}}}_\Lambda{}^{-1}$ (see (108)) is substituted by the diagonals of $\left(\boldsymbol{Q}_\Lambda^{-1} + \boldsymbol{V}_\Lambda \boldsymbol{R}_{\boldsymbol{v}_\Lambda}{}^{-1} \boldsymbol{V}_\Lambda^T\right)^{-1}$ and $\left(\boldsymbol{W}_\Lambda \boldsymbol{R}_{\boldsymbol{w}_\Lambda}{}^{-1} \boldsymbol{W}_\Lambda^T\right)^{-1}$. This yields the *approximate Jacobi* preconditioner $\boldsymbol{M}_\Lambda^{aJ}$ with

$$
\begin{aligned}
\left(\boldsymbol{M}_\Lambda^{aJ}\right)_{i,i'} \;=\; & \left(\sum_s \left(\boldsymbol{Q}_\Lambda^{-1} + \boldsymbol{V}_\Lambda \boldsymbol{R}_{\boldsymbol{v}_\Lambda}{}^{-1} \boldsymbol{V}_\Lambda^T\right)^{-1}_{(s,s)} \boldsymbol{C}^2_{\Lambda,(s,i)} + \right. \\
& \left. \sum_s \left(\boldsymbol{W}_\Lambda \boldsymbol{R}_{\boldsymbol{w}_\Lambda}{}^{-1} \boldsymbol{W}_\Lambda^T\right)^{-1}_{(s,s)} \boldsymbol{X}^2_{\Lambda,(s,i)}\right)^{-1} \delta_{i,i'} \, .
\end{aligned}
\tag{123}
$$

Recall from (108) that the computation of the above diagonal matrices simply amounts to replacing the wavelet finger block matrices in (108) by diagonal matrices. Only 4 very sparse vector multiplications are necessary for each entry to determine $\boldsymbol{M}_\Lambda^{aJ}$.

Other scaling strategies with the same qualitative behavior are conceivable. Nevertheless, the numerical studies in [27] reveal that the above approximate Jacobi preconditioner is clearly superior to the other tested diagonal preconditioners. One could argue about the application of more expensive preconditioners. Nevertheless, as long as an indefinite variant of the KKT–system is considered, incomplete LU or other incomplete factorization preconditioners without pivoting are ruled out because zero entries appear on the diagonal. Pivoting in turn would destroy symmetry. In principle, such techniques apply to the Schur complements $\boldsymbol{\mathcal{S}}_\Lambda$ but require their explicit computation. The same is true for preconditioners based on a sparse approximate inverse. Additionally, the necessary scaling effect is not guaranteed.

6 Stopping Criteria and Final Nested Iteration Algorithm

Recall, in the nested iteration process [35] the intermediate systems have to be solved only within some accuracy tolerance which corresponds to the current discretization error. Thus when using the previous coarse level solution as initial guess for an iteration on the next higher discretization level the current error has to be reduced only by *a constant error reduction factor* α whose size will be discussed below. Moreover, a projection of the initial guess into the next finer trial space is not necessary because one only has to set the additional wavelet coefficients to be zero. Moreover, the matrices of the previous level only have to be extended but not reassembled.

In view of (110), the maximal number of iterations max_{it} on each level needed to realize the error reduction rate α depends on the condition number of the preconditioned Schur complement. Since we have seen that the wavelet based discretization allows us to keep the condition numbers $\kappa_2 = cond(\boldsymbol{M}_\Lambda^{1/2}\boldsymbol{\mathcal{S}}_\Lambda\boldsymbol{M}_\Lambda^{1/2})$ uniformly bounded independently of the scale (see (118)), the maximal number of iterations on each level max_{it} required by a fixed error reduction

$$||\boldsymbol{x} - \boldsymbol{x}_{\Lambda_\ell}||_{\boldsymbol{\mathcal{S}}} \leq \alpha ||\boldsymbol{x} - \boldsymbol{x}_{\Lambda_{\ell-1}}||_{\boldsymbol{\mathcal{S}}} \tag{124}$$

is *independent* of the discretization level. Hence, denoting by $\tilde{\boldsymbol{\Upsilon}}_{\Lambda^\ell}$ and $\tilde{\boldsymbol{\Upsilon}}_{\Lambda^\ell}^{(k)}$ the exact solution of the system (99), respectively the kth iterate and recalling that these arrays are the wavelet coefficients of the respective states $\boldsymbol{x}_{\Lambda_\ell}$, $\boldsymbol{x}_{\Lambda_\ell}^{(k)}$, we infer from (112) that it suffices to choose the number of iterations max_{it} so as to guarantee that

$$||\tilde{\boldsymbol{\Upsilon}}_{\Lambda^\ell} - \tilde{\boldsymbol{\Upsilon}}_{\Lambda^\ell}^{(k)}||_{\boldsymbol{\mathcal{S}}_{\Lambda_\ell}} \leq \alpha/\bar{c}\, ||\tilde{\boldsymbol{\Upsilon}}_{\Lambda^\ell} - \tilde{\boldsymbol{\Upsilon}}_{\Lambda^\ell}^{(0)}||_{\boldsymbol{\mathcal{S}}_{\Lambda_\ell}}, \tag{125}$$

where $\tilde{\boldsymbol{\Upsilon}}_{\Lambda^\ell}^{(0)}$ is obtained by appending zero entries for $\Lambda_\ell \setminus \Lambda_{\ell-1}$ to the approximate solution $\tilde{\boldsymbol{\Upsilon}}_{\Lambda_{\ell-1}}$ from the previous level. Now, there are two ways to guarantee the validity of (125). First, we can conclude from (112) and (125) that at most

$$max_{it} \approx log\,(2\bar{c}/\alpha)\,/\,(-log\,((\sqrt{\kappa_2}-1)/(\sqrt{\kappa_2}+1))) \tag{126}$$

iterations are needed. The second possibility is to use the estimation by the ℓ_2–norm of the residuals of the preconditioned system

$$||\tilde{\mathbf{\Upsilon}}_{\Lambda^\ell} - \tilde{\mathbf{\Upsilon}}^{(k)}_{\Lambda^\ell}||_{\boldsymbol{\mathcal{S}}_{\boldsymbol{\Lambda^\ell}}} / ||\tilde{\mathbf{\Upsilon}}_{\Lambda^\ell} - \tilde{\mathbf{\Upsilon}}^{(0)}_{\Lambda^\ell}||_{\boldsymbol{\mathcal{S}}_{\boldsymbol{\Lambda^\ell}}} \leq \sqrt{\kappa_2}\, ||\boldsymbol{M}^{1/2}_{\Lambda} res^{(k)}_{\Lambda_\ell}||_{\ell_2} / ||\boldsymbol{M}^{1/2}_{\Lambda} res^{(0)}_{\Lambda_\ell}||_{\ell_2} \quad (127)$$

with $res^{(k)}_{\Lambda_\ell} = \boldsymbol{\mathcal{S}}_{\Lambda^\ell}(\tilde{\mathbf{\Upsilon}}_{\Lambda^\ell} - \tilde{\mathbf{\Upsilon}}^{(k)}_{\Lambda^\ell})$. Consequently, we obtain the stopping criteria

$$||\boldsymbol{M}^{1/2}_{\Lambda} res^{(k)}_{\Lambda_\ell}||_{\ell_2} / ||\boldsymbol{M}^{1/2}_{\Lambda} res^{(0)}_{\Lambda_\ell}||_{\ell_2} \leq \alpha/(\bar{c}\sqrt{\kappa_2}) =: \alpha_{rel}\,. \quad (128)$$

In general the exact error reduction factor α is not computable. However, in our particular situation α can be estimated by the following considerations. Since piecewise linear functions are used for the discretization of the states, the discretization error in the L_2–norm for a stepsize h is at best of the order h^2 for smooth functions. Therefore the discrete problem has to be solved only within that accuracy. Due to the differentiation involved in $\tilde{\boldsymbol{\mathcal{B}}}$, the approximation rate with respect to the energy–norm induced by the (infinite dimensional) Schur complement operator $\boldsymbol{\mathcal{S}} = \tilde{\boldsymbol{\mathcal{B}}}\tilde{\boldsymbol{\mathcal{A}}}^{-1}\tilde{\boldsymbol{\mathcal{B}}}^T$ is then only of order one. For example, if we progress from one scale j to the next higher scale $j+1$, i.e. by mesh bisection, the discretization error is expected to decrease at most by a factor $1/4$ in the L_2–norm and by $1/2$ in the energy–norm.

Hence, we fix α for scalewise refinement to $\alpha = 1/2$. Moreover, to determine α_{rel} and max_{it} an upper bound for κ_2 has to be estimated. Choosing $\alpha_{rel} = 10^{-3}$ for all j would cover condition numbers κ_2 up to 1700. Nevertheless, to play safe we set $\alpha_{rel} = 10^{-4}$. Moreover, since max_{it} predicts the worst case we mostly use $max_{it} = 20$ as a pragmatic choice, which rigorously covers only condition numbers up to around 100. This pragmatic choice is motivated by the typical superconvergent behavior of the pcg–method [9, 134]. This superconvergence has been observed in all numerical experiments due to the clustering of the eigenvalues corresponding to the preconditioned systems with only a few small eigenvalues as outliers.

The algorithm for solving a sequence of systems (104) can then be summarized as follows:

set $\ell := \ell_0$ and solve (104) up to machine precision;

- set $\ell := \ell + 1$;
- update the representation of the De noised signals and the control functions from the previous level;
- extend the discretization matrices of the previous level to the current level;
- use the approximate solution of level $\ell - 1$ as initial guess $\tilde{\mathbf{\Upsilon}}^{(0)}_{\Lambda^\ell} \longleftarrow \tilde{\mathbf{\Upsilon}}^{(k)}_{\Lambda_{\ell-1}}$;

- apply the approximate Jacobi preconditioner $\boldsymbol{M}^{aJ}_{\Lambda_\ell}$;
- iterate the simplified Uzawa algorithm ($k := k+1$) until
  a.) the relative residual error in the ℓ_2–norm is below α_{rel} or
  b.) the maximal iteration number $k = max_{it}$ is reached.
- go to the first step.

We illustrate the performance of this scheme by the following simple example and uniform mesh bisection, and refer for further studies to Part 2 [27]. We choose $n_x = n_y = 4$. Furthermore, we allow model uncertainties in all state equations (85), i.e. $\boldsymbol{W} = \boldsymbol{I}$, but not in the output equations (86). This case is realized in our numerical tests with $n_v = n_y$, $\boldsymbol{R}_v = \boldsymbol{I}$ and $\boldsymbol{V} = \boldsymbol{0}$ which forces $\boldsymbol{v}$ to vanish. The model matrices

$$\boldsymbol{A} = \begin{pmatrix} -1 & & & \\ -10 & -1 & & \\ & -1 & -1 & \\ & & -10 & -1 \end{pmatrix}, \quad \boldsymbol{B} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \boldsymbol{C} = \boldsymbol{I}$$

are normalized with respect to the L_2–norm using the reference solutions $\boldsymbol{x}$ and $\boldsymbol{y}$. These reference solutions are determined by solving the model (85) and (86) with $\boldsymbol{w} = 0$, $\boldsymbol{x}(0) = \mathbf{1}$ and a piecewise linear control function u, oscillating between 10 and -10 with $u(0) = 10$ and step size 0.25. As regularization matrices we choose $\boldsymbol{Q} = \boldsymbol{I}$ as well as $\boldsymbol{R}_w = \boldsymbol{I}$.

The condition numbers of the Schur complements and their preconditioned variants are recorded in Table 1 along with the iteration numbers max_{it} according to (126) obtained for $\alpha = 1/2$ and $\boldsymbol{M}_\Lambda = \boldsymbol{M}^{aJ}_\Lambda$. Obviously the condition numbers

Table 1: cond $\left(\boldsymbol{M}_\Lambda^{1/2}\boldsymbol{\mathcal{S}}_\Lambda\boldsymbol{M}_\Lambda^{1/2}\right)$.

Scale	$\boldsymbol{M}_\Lambda = \boldsymbol{I}$	$\boldsymbol{M}_\Lambda = \boldsymbol{M}^{aJ}_\Lambda$	max_{it}
1	5.88e+01	49.2	14
2	1.32e+02	55.5	15
3	4.46e+02	60.3	15
4	1.64e+03	63.3	16
5	6.33e+03	65.4	16
6	2.48e+04	67.0	16
7	9.87e+04	68.2	16
8	3.93e+05	69.0	17
9	1.57e+06	69.7	17

$cond\,(\boldsymbol{\mathcal{S}}_\Lambda)$ increase roughly by a factor 4 in each refinement step. In contrast, one observes the predicted boundedness of the condition number $cond\left(\boldsymbol{M}_\Lambda^{1/2}\boldsymbol{\mathcal{S}}_\Lambda\boldsymbol{M}_\Lambda^{1/2}\right)$.

Table 2 displays the following information on the above nested iteration algorithm: the number of iterations k on scale j, the error $e_j = ||\boldsymbol{d}^k_{\boldsymbol{x}_{\Lambda_j}} - \boldsymbol{d}^{ref}_{\boldsymbol{x}}||_{\ell_2}$ between the wavelet coefficients of the obtained approximation and the reference solution as well as the quotient e_{j-1}/e_j where j denotes the current scale. We measure here the error in the ℓ_2–norm instead of the energy–norm. According to the above discussion, we expect a quotient up to 4. Moreover, to provide an idea of the size of the problems we include also the size dim of $\boldsymbol{\mathcal{S}}_\Lambda$, their non zero entries $nz(\boldsymbol{\mathcal{S}}_\Lambda)$ and the residuals $\|res\|_{\ell_2} = \|\boldsymbol{\mathcal{S}}_\Lambda \boldsymbol{d}^{appr}_{\boldsymbol{x}_\Lambda} - \boldsymbol{rhs}_\Lambda\|_{\ell_2}$ on each scale, where $\boldsymbol{rhs}_\Lambda$ denotes the right hand side of (114). The last line gives the results for solving the system only on scale 9 with $\mathbf{0}$ as initial guess in the simplified Uzawa algorithm.

Table 2: Results corresponding to nested iteration.

Scale j	k	e_j	e_{j-1}/e_j	dim	$nz(\boldsymbol{\mathcal{S}}_\Lambda)$	$\|res\|_{\ell_2}$
1	16	9.4525e−01		12	70	6.0166e−15
2	17	8.8248e−01	1.07	20	230	4.0636e−04
3	17	2.4626e−01	3.58	36	730	4.9402e−03
4	17	6.6437e−02	3.70	68	2264	3.4785e−03
5	17	1.7124e−02	3.87	132	6776	2.6048e−03
6	17	4.3262e−03	3.95	260	18846	2.3966e−03
7	15	1.0811e−03	4.00	516	48556	2.2025e−03
8	14	2.6419e−04	4.09	1028	118454	2.5533e−03
9	10	6.3861e−05	4.13	2052	276688	2.1355e−03
9	34	5.8380e−05		2052	276688	2.4164e−03

Due to the geometric progression of the problem sizes we see that the computational cost for nested iteration up to scale 9 corresponds to 20 or 21 iterations of the simplified Uzawa algorithm applied to the system on scale 9 only. Perhaps more important than the expected higher efficiency of nested iteration is the fact that it provides successively improved approximations at a much earlier stage. We can see also that it is not necessary to determine on each scale the exact solution. The best approximation error is closely reached, although the residuals are still quite large and in full agreement with the size of the condition numbers in this example. The quotient e_{j-1}/e_j is as good as an exact solver can provide to produce meaningful results associated to discretization accuracy.

7 Conclusions

In order to meet the on–line requirements of state estimation, i.e. to offer at any instant of time an approximate solution whose quality increases in time, we have proposed a nested iteration scheme based on a hierarchy of wavelet Galerkin discretizations. For each discretization level we exploit extensively the structure of the underlying system of equations for deriving a well tailored iterative scheme termed simplified Uzawa algorithm. The number of iterations needed on each level to reduce the current error by only a fixed factor in the context of nested iteration depends on the condition numbers. The condition numbers can be kept bounded with the aid of wavelet concepts using an inexpensive diagonal preconditioner, namely an approximate Jacobi preconditioner. Hence, we can conclude that the presented iteration scheme provides at any instant of time an improved state estimation. Moreover, depending on the system inherent features, the nested iteration process can be more efficient than solving the discrete problem directly for the highest level of resolution.

However, in spite of the accomplished asymptotic boundedness, large condition numbers may still arise due to system inherent features as for example, a poor observability measure, stiff differential equations or inadequate regularization. These issues will be addressed in more detail in Part 2 for more challenging test examples. In particular, the interplay between preconditioning and regularization will be studied there.

The extension of the proposed algorithm to nonlinear systems with inequality constraints is possible. In particular, the linear systems arising when active set methods are employed, have the form (98) and the introduced elimination process is not affected by the nonlinearity. However, although the condition numbers will be still bounded independently of the refinement, quantitative bounds are not clear.

Moreover we would like to mention, that the hierarchy of problem discretizations can also be exploited for the purpose of regularization in connection with noisy data. On one hand, regularization by discretization can be realized by determining an appropriate adaptively refined discretization level [29]. On the other hand, level depending regularization parameters can be incorporated. For example, in [128] a related discussion on wavelet accelerated Tikhonov regularization is given under the assumption of an a priori known noise level.

Last but not least it should be noted that the numerical concept carries over in a straight forward manner to the solution of real–time optimal control problems as they arise in model predictive control applications [139].

Iterative Algorithms for Multiscale State Estimation, Part 2: Numerical Investigation[4]

Thomas Binder, Luise Blank, Wolfgang Dahmen, Wolfgang Marquardt

in: Journal of Optimization Theory and Applications (JOTA), Vol. 111 (3), 529-551, 2001.

Abstract

Usually direct methods are employed for the solution of linear systems arising in the context of optimization. However, motivated by the potential of multiscale refinement schemes for large problems of dynamic state estimation, we investigate in this paper the application of iterative solvers based on concepts developed in [26]. Specifically, we explore the effect of different system reductions for various Krylov–space iteration methods as well as three concepts of preconditioning. The first one is the normalization of states and outputs, which also favors error analysis. Secondly, diagonal, scale dependent preconditioners are compared. They all bound the condition numbers independently of the refinement scale, but exhibit significant quantitative differences. Finally, the effect of the regularization parameter on the condition and iteration numbers is analyzed. It turns out that a so called simplified Uzawa scheme with Jacobi preconditioning and suitable regularization parameter is most efficient. However, the experiments also reveal that further improvements are necessary.

Key Words: Dynamic optimization, optimal control, iterative linear algebra, wavelets, preconditioning.

1 Introduction

Due to the on–line requirements of process monitoring, multiscale state estimation aims at providing an approximation of a dynamic system at any instant of time. This can be achieved by providing a coarse approximation after a very short period and, in the framework of nested iteration, by upgrading this approximation through

[4]This work has been supported by the "Deutsche Forschungsgemeinschaft" under grant no. MA1188/6

a hierarchy of successively refined optimization problems [20, 26]. In order to fully exploit prior calculations in the next solution step and observing the fact that it is sufficient to solve each optimization problem only up to the discretization error accuracy, iterative solvers suggest themselves. However, since iterative techniques have to date apparently not received much attention in the context of state estimation and/or control on receding horizons (see [6] for an introduction), the objective of this paper is to investigate several iterative schemes in the context of multiscale state estimation.

The outline of the paper is as follows: In Section 2 we introduce the specific problem formulation considered here. Then, to explain typical obstructions arising in the above context, we briefly recall several measures of observability and apply them to the two model problems described at the end of the section. Section 3 is devoted to the description of a wavelet based multiscale discretization for problems of this type. In Section 4 we present extensive numerical experiments. Specifically we explore the influence of several system reductions on the performance of different iterative schemes, discuss the effect of certain state normalizations as well as the interplay of the choice of regularization parameters with a tailored preconditioned Uzawa scheme introduced in Part 1 [26]. We conclude with a brief summary of our findings in Section 5.

2 Problem Formulation

The dynamic state estimation can be formulated as a constrained optimization problem, where the difference between measured data $\boldsymbol{z}$ and modeled outputs $\boldsymbol{y}$ is to be minimized [20, 26]. The model equations including in addition to $\boldsymbol{y}$ the states $\boldsymbol{x}$ and the known controls $\boldsymbol{u}$ form the constraints. Moreover, model error functions $\boldsymbol{v}$ and $\boldsymbol{w}$ are included in the model equations to represent unstructured model uncertainty. They have to be estimated together with states and outputs. We stress that the problem in its original form is an ill–posed inverse problem. Therefore some regularization is necessary [29].

Since we concentrate on the solution of the final linear systems, we consider here only linear model problems without inequality constraints. Moreover, while our general approach of state estimation is based on a moving horizon approach, we fix the horizon in this paper and scale it to $[0, 1]$. Furthermore, in the chosen examples we allow model uncertainties in all model equations which possibly leads to overparameterization and, therefore, to higher problem complexity. Motivated by the Schur complement considerations in Part 1 [26], we do not distinguish between the regularizations of $\boldsymbol{y} - \boldsymbol{z}$ and $\boldsymbol{v}$, but between their regularization and the one of $\boldsymbol{w}$. Hence, we focus here on the following particular setting, where we suppress for convenience explicit references to the domain $[0, 1]$.

For given $\boldsymbol{z} \in (L_2)^{n_y}$ and $\boldsymbol{u} \in (L_2)^{n_u}$ and unknown functions $\boldsymbol{x} \in (H^1)^{n_x}$, $\boldsymbol{w} \in (L_2)^{n_x}$ and $\boldsymbol{y}, \boldsymbol{v} \in (L_2)^{n_y}$ the resulting optimization problem has the form

$$\min_{\boldsymbol{x}_0, \boldsymbol{v}, \boldsymbol{w}} \left\{ ||\boldsymbol{y} - \boldsymbol{z}||_{L_2^{n_y}} + ||\boldsymbol{v}||_{L_2^{n_y}} + \varepsilon \, ||\boldsymbol{w}||_{L_2^{n_x}} \right\} \tag{129}$$

subject to the constraints

$$\begin{aligned} \dot{\boldsymbol{x}} - \boldsymbol{A}\boldsymbol{x} - \boldsymbol{w} &= \boldsymbol{B}\boldsymbol{u} \,, & (130) \\ \boldsymbol{y} - \boldsymbol{C}\boldsymbol{x} - \boldsymbol{v} &= \boldsymbol{0} \,. & (131) \end{aligned}$$

The size of the problem is dominated by the number of states n_x. Note that in the above regularized problem the initial data $\boldsymbol{x}(0)$ are not given but have to be estimated. In addition, the observability measure of the model represented by the constraints, the stiffness of the differential equations, the incorporated model error functions and the choice of the regularization parameter ε influence the condition of the formulated optimization problem and hence the numerical treatment of the resulting linear systems of equations. We briefly recall now the notion of observability.

2.1 Observability and Observability Measures

The linear system (130)–(131) is called observable if and only if for fixed $\boldsymbol{u}, \boldsymbol{w}, \boldsymbol{v}$ we can uniquely reconstruct the state $\boldsymbol{x}$ and their initial values from given outputs $\boldsymbol{y}$. The *observability measure* quantifies the sensitivity of $\boldsymbol{y}$ with respect to variations in $\boldsymbol{x}(0)$. Obviously, the size of the observability measure influences the condition of the optimization problem. Different measures have been proposed in the literature. Since it is not clear yet which observability measure is most adequate for our setting we apply the most commonly used measures, described below, to the test examples. For a more thorough treatment we refer to [110, 185].

In principle, a reconstruction of $\boldsymbol{x}$ is obtained by subsequent differentiation of $\boldsymbol{y}$ and repeated substitution of (130) into the differentiated equation (131) up to degree $n_x - 1$. This leads one to define

$$O_1^{\rho} \;:=\; [\boldsymbol{C}^T, \boldsymbol{A}^T\boldsymbol{C}^T, (\boldsymbol{A}^T)^2\boldsymbol{C}^T, \ldots, (\boldsymbol{A}^T)^{\rho-1}\boldsymbol{C}^T]^T \quad \rho = 1, \ldots, n_x, \tag{132}$$

and to call the linear system (130)–(131) observable if and only if the matrix $O_1^{n_x}$ is of rank n_x. The Kalman observability index ρ refers to the smallest integer needed to ensure full rank. The *Kalman observability matrix measure* is defined by the smallest singular value $\sigma_{min}(O_1^{\rho})$ of the observability matrix O_1^{ρ}.

Another way to measure observability is given by the Popov–Belevitch–Hautus rank test. Let λ_i denote the eigenvalues of $\boldsymbol{A}$ and

$$O_2(\lambda_i) \;:=\; [\lambda_i \boldsymbol{I} - \boldsymbol{A}^T, \boldsymbol{C}^T]^T. \tag{133}$$

Then, the pair $(\boldsymbol{A}, \boldsymbol{C})$ is observable if and only if $rank[O_2(\lambda_i)] = n_x$ for all $\lambda_i,\ i = 1, \ldots, n_x$. As before, the observability measure is defined by the smallest singular value $\bar{\sigma}_{min}(O_2) := \min_{\lambda_i, i=1,\ldots,n_x}\{\sigma_{min}(O_2(\lambda_i))\}$.

The observability Gramian is defined by

$$O_3 := \int_0^\infty e^{\boldsymbol{A}^T\tau}\boldsymbol{C}^T\boldsymbol{C}e^{\boldsymbol{A}\tau}d\tau \tag{134}$$

and involves the fundamental solution $e^{\boldsymbol{A}t}$ of the linear system (130)–(131). Observability is given if $\text{rank}(O_3) = n_x$. The associated observability measure is $\sigma_{min}(O_3)$. Note that the observability Gramian is only computable for strictly stable systems, i.e. all eigenvalues λ_i of $\boldsymbol{A}$ have to have negative real part.

The modal transformation based measure is based on the observability test

$$\kappa_i \ := \ \left(\bar{\boldsymbol{m}}_i^T\boldsymbol{C}^T\boldsymbol{C}\boldsymbol{m}_i\right) / \left(\bar{\boldsymbol{m}}_i^T\boldsymbol{m}_i\right) > 0 \quad \text{for all eigenvectors } \boldsymbol{m}_i \text{ of } \boldsymbol{A}$$

where $\bar{\boldsymbol{m}}_i$ denotes the complex conjugate of $\boldsymbol{m}_i$. The associated observability measure is given by $\kappa_{min} = \min_i\{\kappa_i\}$.

2.2 Test Problems

We describe next two small test problems which, however, exhibit the essential obstructions one encounters in connection with large realistic problems.

Example 2.1: In the first example the input is a given control u which is subsequently integrated four times. The result is the output of the system. Amplification factors between the stages and the feedbacks appear in the model equations as constants $k_i \in R$ and $c_i \in R$ respectively. Then the resulting system matrices are given by

$$\boldsymbol{A} = \begin{pmatrix} c_1 & 0 & 0 & 0 \\ k_1 & c_2 & 0 & 0 \\ 0 & k_2 & c_3 & 0 \\ 0 & 0 & k_3 & c_4 \end{pmatrix} \quad \boldsymbol{B} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad \boldsymbol{C}^T = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}. \tag{135}$$

Model correction terms are permitted in all equations. Hence, uniqueness of the estimation problem can only be expected for a suitable regularization. As a case study we choose the constants $\boldsymbol{k} = (5, 10, 5)^T$, $\boldsymbol{c} = -(20, 10, 0.5, 5)^T$. The control function u is chosen as a piecewise linear function, oscillating between 10 and -10 with $u(0) = 10$ and step size 0.25. Moreover, we use initial data $\boldsymbol{x}(0) = (1, 0, 0.5, -0.5)^T$ and $\boldsymbol{w} = \boldsymbol{0}$, $\boldsymbol{v} = \boldsymbol{0}$ to generate the nominal solution. As will be seen, despite its very simple structure and low dimension, this example might exhibit typical system properties such as small observability measures, which is typical for industrial applications.

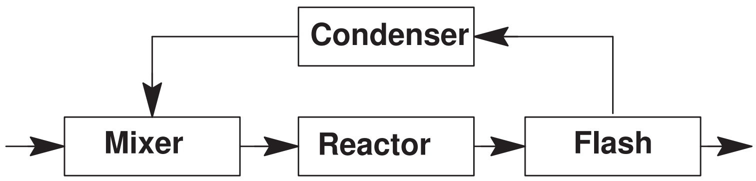


Figure 24: Glycol process.

Example 2.2: As a second test problem we consider a simple model of a chemical process which produces ethylene glycol from ethylene oxide and glycol (Figure 24). The nonlinear model lumps a number of related states and outputs. Further simplification is accomplished by linearization at some stationary operating point and by subsequent elimination of the algebraic state variables to result in a linear model with twelve states representing the concentrations of the five major chemical components, the temperature and the holdup in the reactor as well as in the flash. There are five measurements which comprise the temperatures and the levels in the reactor and in the flash as well as the pressure in the flash. The nominal solutions are generated from nominal zero initial conditions and given forcing functions. The forcing functions comprise flow rates within the process as well as heating and cooling duties in the reactor and the flash. For brevity we do not give the numerical values of the system matrices. Details on the model can be found in [21]. However, the eigenvalues λ_i provide insight into system behavior and the values of the unscaled model are given by $\{-1.9e2, -1.6e2, -3.8, -2.3, -1.0 \pm i\,0.12, -1.1, -1.97, -1.94, -0.89, -6.1e-6, 0\}$. To ensure that relative errors are equally weighted, the model matrices are scaled such that the components of the nominal solution are roughly of unit size in L_2. The reduction of the differential algebraic equations to a model of type (130) by elimination of the algebraic variables has to be paid by the loss of the typically very sparse structure in the model matrices as depicted Figure 25. Moreover, the

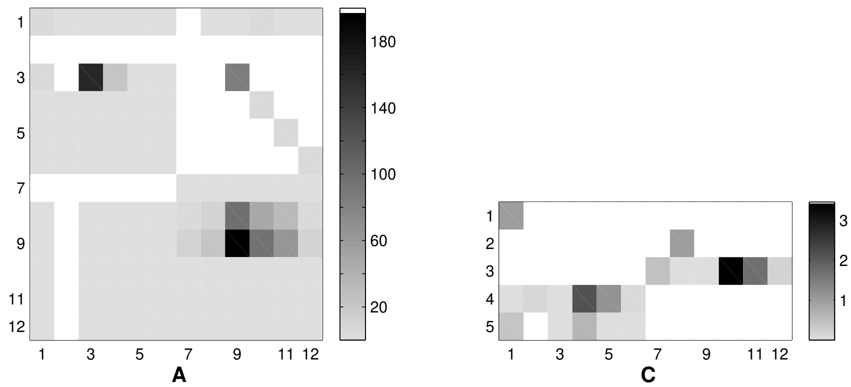


Figure 25: Model matrices.

grey scale in the sparsity patterns of $\boldsymbol{A}$ and $\boldsymbol{C}$ indicate the absolute values of the matrix entries after normalization by means of the nominal solution. As in the first example, we incorporate model correction terms in all state and output equations, and deliberately overparameterize the problem.

The observability measures for the test examples are given in Table 3. We also calculated the measures for the special choice $\boldsymbol{C} = \boldsymbol{I}$, which means that all states can be measured directly. This obviously reflects the case of perfect observability and serves as a nominal value for a comparison. The numbers have been calculated for the unscaled problems. Unfortunately, the observability measures applied are not scale invariant. Scaling is crucial for the numerical algorithm and will be discussed in more detail in Section 4.3. The computed numbers somehow reflect the difficulty

Table 3: Observability measures for the test examples.

Test example		Kalman matrix measure $\sigma_{\min}(O_1^\rho)$	Popov–Belevitch Hautus measure $\bar{\sigma}(O_2)$	Gramian based measure $\sigma_{\min}(O_3)$	Modal transformation based measure $\kappa_{\min}$
	ρ				
Example 2.1	4	7.0e-1	7.4e-2	5.5e-6	5.5e-3
perfect observable	1	1	1	2.4e-2	1
Example 2.2	3	3.3e-3	1.2e-4	not applicable	1.9-5
perfect observable	1	1	1	not applicable	1

a numerical scheme is confronted with when reconstructing the states from the measurements. Varying the entries k and c in *Example 2.1*, leads to significantly different observability measures. This indicates that although the example has only four states and one output function the reconstruction can face severe problems. The observability measure for *Example 2.2* is clearly worse than for *Example 2.1* with respect to all measures. It is rather difficult to draw precise quantitative conclusions from the measures since the numbers differ significantly depending on the respective concept. However, the measures can be used to qualitatively appraise the numerical difficulty one has to expect when solving the reconstruction problem.

3 Wavelet–Galerkin Discretization

We discretize the saddle point problem corresponding to the optimization problem (129)–(131) by a wavelet–Galerkin ansatz with a chosen, finite index set Λ of wavelet basis functions. A detailed description of the discretization is given in Part 1 [26]. In addition, we refer readers who are not familiar with wavelets to [43, 56, 64]. We recall here only briefly the main notational ingredients. The functions $\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{y}, \boldsymbol{v}$ are discretized with globally continuous, piecewise linear spline wavelets $\Psi_{j,k}$, $(j,k) \in \Lambda$ ($x_i = \sum_{j,k} d_{j,k}\Psi_{j,k}$). The dual wavelets $\tilde{\Psi}_{j,k}$, $(j,k) \in \Lambda$, form the basis for the

Lagrange multipliers $\boldsymbol{\nu}$ corresponding to the output equation (131). Moreover, the piecewise constant Haar wavelets $\Psi^H_{j,k}$, $(j,k) \in \Lambda'$, are chosen for discretizing $\boldsymbol{w}, \boldsymbol{u}$ and the Lagrange multipliers $\boldsymbol{\mu}$ corresponding to the differential equations (130). Since for a given coarsest level the number of coarse level scaling functions for the Haar basis is by one less than the number of corresponding piecewise linear scaling functions, the index set Λ' differs from Λ only in one scaling function index. In fact, the necessary variation of initial values requires reserving at least one degree of freedom in each state variable x_i in excess of the test conditions. Let us collect now in $\boldsymbol{d_x}$ all wavelet coefficients of the states $\boldsymbol{x}_\Lambda$ ($\boldsymbol{d_x} = \left(\boldsymbol{d}_{x_i}{}^{\Lambda}\right)_{i=1,\ldots n_x}$) and likewise for $\boldsymbol{d_y}, \boldsymbol{d_v}, \boldsymbol{d_w}$ etc. where we have suppressed for convenience the index Λ. Furthermore we use the shorthand notation of $\langle\langle \boldsymbol{\Psi}_\Lambda, \boldsymbol{\Psi}_{\Lambda'} \rangle\rangle = \left(\langle \psi_{(j,k)}, \psi_{(j',k')} \rangle\right)_{(j,k)\in\Lambda,(j',k')\in\Lambda'}$ for L_2–mass matrices, which typically have sparse, so called finger structure. To describe the discretization matrix in a rather condensed form we use the Kronecker product $\boldsymbol{A} \otimes \boldsymbol{B} := (a_{i,j}\boldsymbol{B})_{\substack{i=1,\ldots,n \\ j=1,\ldots,m}}$ of a matrix $\boldsymbol{A} \in R^{n\times m}$ with a matrix $\boldsymbol{B}$. With $\boldsymbol{I}_n$, $\boldsymbol{I}_\Lambda$ we denote the identity matrix of size $n \times n$, respectively $(\#\Lambda) \times (\#\Lambda)$ and define the matrices

$$\begin{aligned}
&\boldsymbol{Q}_\Lambda = \boldsymbol{I}_{n_y} \otimes \langle\langle \boldsymbol{\Psi}_\Lambda, \boldsymbol{\Psi}_\Lambda \rangle\rangle, \quad \boldsymbol{C}_\Lambda = \boldsymbol{C} \otimes \boldsymbol{I}_\Lambda, \quad \boldsymbol{B}_\Lambda = \boldsymbol{B} \otimes \boldsymbol{I}_\Lambda, \\
&\boldsymbol{X}_\Lambda = \boldsymbol{I}_{n_x} \otimes \langle\langle \boldsymbol{\Psi}^H_{\Lambda'}, \dot{\boldsymbol{\Psi}}_\Lambda \rangle\rangle - \boldsymbol{A} \otimes \langle\langle \boldsymbol{\Psi}^H_{\Lambda'}, \boldsymbol{\Psi}_\Lambda \rangle\rangle \,.
\end{aligned}$$

Then, the Galerkin discretization leads to the following linear system of equations

$$\begin{pmatrix}
 & & & & \boldsymbol{X}_\Lambda{}^T & -\boldsymbol{C}_\Lambda{}^T \\
 & 2\boldsymbol{Q}_\Lambda & & & & \boldsymbol{I} \\
 & & 2\boldsymbol{Q}_\Lambda & & & -\boldsymbol{I} \\
 & & & 2\varepsilon\boldsymbol{I} & -\boldsymbol{I} & \\
\boldsymbol{X}_\Lambda & & & -\boldsymbol{I} & & \\
-\boldsymbol{C}_\Lambda & \boldsymbol{I} & -\boldsymbol{I} & & &
\end{pmatrix}
\begin{pmatrix}
\boldsymbol{d_x} \\ \boldsymbol{d_y} \\ \boldsymbol{d_v} \\ \boldsymbol{d_w} \\ \boldsymbol{d_\mu} \\ \boldsymbol{d_\nu}
\end{pmatrix}
=
\begin{pmatrix}
0 \\ 2\boldsymbol{Q}_\Lambda \boldsymbol{d_z} \\ 0 \\ 0 \\ \boldsymbol{B}_\Lambda \boldsymbol{d_u} \\ 0
\end{pmatrix} . \tag{136}$$

Note that all the wavelet stiffness matrices appearing in $\boldsymbol{X}_\Lambda$ are singular, since the number of rows is by one less than the number of columns. Moreover, in general, $\boldsymbol{C}_\Lambda$ is also singular due to the singularity of $\boldsymbol{C}$.

4 Numerical Results

In this section we examine the performance of several iterative methods for the solution of the KKT system (136). *Examples 2.1* and *2.2* serve as test problems.

It will be seen that in particular in *Example 2.2* extremely large condition numbers are encountered. Here, normalization of the states and the judicious choice of the regularization parameter ε in (129) have to complement an efficient preconditioner. Therefore, unless stated otherwise, the experiments are carried out for *Example 2.2* with state normalization and the choice $\varepsilon = 10^{-4}$, which will be seen to be optimal, while for *Example 2.1* we set $\varepsilon = 1$.

While one expects an adaptive selection of trial spaces, that is the adaptive choice of the index sets Λ to be important for real time computation [21], we confine our tests of the iterative solution of the KKT–matrix to uniform discretizations of mesh size 2^{-J}, i.e. the index set Λ_J consists of all possible wavelet indices (j,k) up to scale J. For nonuniform meshes we expect similar results. Nevertheless, since the linear system (136) is part of a multiscale approach (see Part 1), we have to consider mesh refinement and its effect on the iteration numbers. The objective is to keep the number of required iterations low and uniformly bounded with increasing scale J. The choice of the coarsest scale depends, among other things, on the number of outputs relative to the number of states. For *Example 2.2* we can start with scale 1, while scale 2 is necessary for *Example 2.1*. As maximal scale we choose 7 in both examples which corresponds to the mesh size 2^{-7}. Of course, the corresponding systems have different dimensions, which should be taken into account when interpreting the iteration numbers.

Due to the nested iteration ansatz in the multiscale approach (see Part 1), we use the approximate solution from the preceding scale $j-1$ as initial guess for the current discretization with Λ_j. Here the hierarchical structure of wavelet bases implies, that one only has to append zero coefficients to the array of wavelet coefficients from the previous solution.

A typical stopping criterion requires that the residual $\boldsymbol{res} = \boldsymbol{K} \cdot \boldsymbol{sd} - \boldsymbol{rhs}$ is smaller than some target tolerance tol unless some maximal number max_{it} is exceeded. Here $\boldsymbol{K}$ is the (preconditioned) system matrix under consideration and $\boldsymbol{sd}, \boldsymbol{rhs}$ denote the computed approximate solution and right hand side, respectively. Of course, these quantities depend on the particular preconditioner $\boldsymbol{M}_\Lambda$, on the current index set Λ and on the regularization parameter ε. However, due to the large condition numbers, the residuals are not very reliable. Therefore we monitor here for testing purposes the approximation to the exact error $\boldsymbol{e}_{it} = \boldsymbol{sd}_{it} - \boldsymbol{sd}_{dir}$, the difference between the iterative solution $\boldsymbol{sd}_{it}$ and the solution $\boldsymbol{sd}_{dir}$ which for *Example 2.1* is produced with the aid of a direct solver. Because of the extremely high condition numbers in *Example 2.2* even the results of a direct solver are too inaccurate. Therefore $\boldsymbol{sd}_{dir}$ is in this case chosen as the solution produced by the routine *symmlq*[5] until the residual is of the order of machine accuracy. We are primarily interested in the L_2 error of the states. In view of the Riesz basis property [56, 64], this simply amounts to taking the Euclidean norm $||\boldsymbol{e}_{it}||_{\ell_2}$. Since our primary interest at this stage is not a time optimal complete code but rather a possibly thorough answer to

[5]The version *symmlq* employed here has been kindly provided by Michael Saunders.

the questions stated in the introduction, we deviate in this paper from the original nested iteration concept suggested in Part 1 and iterate on each level until a certain fixed accuracy is attained. Specifically, we choose the stopping criteria

$$||\boldsymbol{e}_{it}||_{\ell_2} \leq tol, \qquad k = max_{it}$$

with $max_{it} = 3000$ and $tol = 10^{-6}$ for *Example 2.1* and $tol = 10^{-4}$ for *Example 2.2*. Moreover, we choose Matlab (Version 5.2) as a convenient computational environment, and partly include C and C++ codes. For a description of the iterative methods used below we refer to [14, 82] and the references cited there.

4.1 Comparison of Different System Formulations

As a first step for analyzing the iterative solvers we investigate possible inexpensive system reductions. Specifically, we denote the KKT system (136) by red_1. The elimination of $\boldsymbol{d_y}, \boldsymbol{d_v}, \boldsymbol{d_w}$ with their representation given by the Lagrange parameters $\boldsymbol{d_\mu}$ and $\boldsymbol{d_\nu}$ yields the system red_2. Additional elimination of $\boldsymbol{d_\nu}$ gives the system red_3 which corresponds to the variables $\boldsymbol{d_x}$ and $\boldsymbol{d_\mu}$. The last reduction step (red_4) eliminates $\boldsymbol{d_\mu}$ and supplies a positive definite matrix, which is the Schur complement $\boldsymbol{\mathcal{S}}_\Lambda$ of red_2

$$\boldsymbol{\mathcal{S}}_\Lambda = \boldsymbol{C}_\Lambda^T \left(\boldsymbol{I}_{n_y} \otimes \langle\langle \boldsymbol{\Psi}_{\Lambda_y}, \boldsymbol{\Psi}_{\Lambda_y} \rangle\rangle \right) \boldsymbol{C}_\Lambda + 2\varepsilon \boldsymbol{X}_\Lambda^T \boldsymbol{X}_\Lambda \, . \tag{137}$$

The different formulations of the KKT system (136) exhibit different features, as for example condition numbers, dimensions and definiteness. Hence, we investigate first the influence of these differences on the performance of an iterative scheme. In particular, since the number of nonzero entries remains roughly the same for all reductions, we compare the numbers of iterations. As an iterative solver we choose *symmlq* because it is designed for symmetric not necessarily definite systems and therefore applies to each formulation. Like all iterative methods tested below it is based on the construction of orthogonal bases for Krylov subspaces. In the case of *symmlq* a three term recurrence relation is used [14, 149]. We use the diagonal, left hand side preconditioner $\boldsymbol{M}^{l_2}$ with the l_2–norm of the corresponding matrix rows as entries. The justification for this choice which can be applied to all system formulations will be given later in Section 4.4. In the Tables 4 and 5 the iteration numbers corresponding to the above stopping criteria are recorded for both test examples.

Obviously, even though the reduced systems may have larger condition numbers (e.g. tests have confirmed that $\mathrm{cond}_2(\mathrm{red}_4) \approx (\mathrm{cond}_2(\mathrm{red}_1))^2$), the iteration numbers clearly indicate, that the use of the smaller system is preferable. On the other hand, it would be desirable not to assemble the Schur complement $\boldsymbol{\mathcal{S}}_\Lambda$.

4.2 Comparison of Different Iterative Methods

Exploiting the specific problem structure we developed in Part 1 a *simplified Uzawa* scheme. Essentially it is based on the *pcg*–method applied to the Schur complement

Table 4: Iteration numbers for different system formulations, *Example 2.1.*

Scale	red_1	red_2	red_3	red_4
2	64	43	36	21
3	103	80	70	38
4	138	112	94	53
5	163	137	112	62
6	177	161	132	73
7	201	182	155	85

Table 5: Iteration numbers for different system formulations, *Example 2.2.*

Scale	red_2	red_4
1	492	130
2	906	224
3	1432	359
4	2192	463
5	2772	555
6	max	578
7	max	596

$\boldsymbol{\mathcal{S}}_\Lambda$ but uses the system red_2. However, the difference is that the *simplified Uzawa* algorithm calculates the solution $\boldsymbol{d_x}, \boldsymbol{d_\mu}$ and $\boldsymbol{d_\nu}$ iteratively without setting up $\boldsymbol{\mathcal{S}}_\Lambda$ and at nearly the same cost as *pcg* needs for calculating only $\boldsymbol{d_x}$. Hence it is not surprising that the respective numbers of iterations are the same as all tests confirm (not shown here).

Therefore, to compare the performance of different iterative methods with the *simplified Uzawa* scheme we choose the system red_2. As left hand side preconditioner we use again $\boldsymbol{M}^{l_2}$. In addition to *symmlq* we apply *gmres(k)* and *bicg* although they do not exploit symmetry as well as the *simplified Uzawa* scheme. Methods like *cgne* and *cgnr* are inappropriate since they square the already high condition numbers. The advantage of using *symmlq, bicg* or *gmres* could be, that they apply conceptually to matrices whose condition numbers are smaller than the condition numbers of the Schur complements, which are relevant for the simplified Uzawa algorithm. It is remarkable that *gmres* only converged without restart, to which all listed results refer to. In view of the enormous cost, this method turns out not to be suitable.

The Tables 6 and 7 show that *symmlq* and *bicg* require nearly the same number of iterations. Nevertheless, the computational work for one iteration in the *bicg* method roughly doubles that required by *symmlq,* since *bicg* biorthogonalizes two sequences. For *Example 2.1 gmres* without restart is slowest, while the *simplified Uzawa* algorithm yields the best results. For *Example 2.2*, *gmres* requires the lowest

Table 6: Iteration numbers for different iterative methods, *Example 2.1*.

Scale	symmlq	bicg	simplified Uzawa	gmres
2	43	44	23	41
3	80	80	40	77
4	112	112	64	149
5	137	137	90	288
6	161	163	123	527
7	182	182	166	927

Table 7: Iteration numbers for different iterative methods, *Example 2.2*.

Scale	symmlq	bicg	simplified Uzawa	gmres
1	492	488	146	75
2	906	877	263	133
3	1432	1580	404	249
4	2192	2427	613	478
5	2772	2963	709	899
6	max	max	743	–
7	max	max	689	–

iteration numbers for the lower scales. However, since *gmres* employs orthogonalization with respect to all previous directions, significantly more work is needed for one iteration when progressing to higher scales. For scale 6 and 7, the required storage is even too large to allow simple testing.

Overall, the *simplified Uzawa* algorithm seems to perform best. The above comparison of different iterative schemes motivates now to explore different ways of improving the efficiency of the simplified Uzawa algorithm. Since for this method the condition number and the eigenvalue distribution of $\boldsymbol{\mathcal{S}}_\Lambda$ is as relevant as for the corresponding pcg–method, we aim at decreasing the condition numbers. Three possible major influences are analyzed in the following: normalizing the states and outputs, preconditioning $\boldsymbol{\mathcal{S}}_\Lambda$ and varying the parameter ε.

4.3 Normalization

Normalizing the states, outputs and measurements using possibly available information about the nominal solution can also be viewed as preconditioning with an additional change of the previously chosen regularization. However, normalized variables ($||x_i||_{L_2} \approx 1$ etc.) should be introduced before regularization is fixed. The fact that normalization can have a significant effect on condition numbers is illustrated by Table 8 which shows the original and normalized system for *Example 2.2* with $\varepsilon = 1$.

It reveals that proper normalization is necessary for this example. Aside from the effect on conditioning the system, normalization also weighs the state errors equally, which is advantageous for the error analysis.

Table 8: cond($\boldsymbol{\mathcal{S}}_\Lambda$) without and with normalizing of $\boldsymbol{x}$ and $\boldsymbol{y}$, *Example 2.2.*

Scale	without	with
1	8.1694e+19	9.2508e+10
2	1.4066e+20	1.0107e+11
3	1.2968e+19	5.4481e+10
4	3.8306e+19	3.1693e+10
5	6.4281e+19	1.9665e+10
6	1.0396e+20	1.4867e+10
7	3.2429e+19	2.2272e+10

4.4 Comparison of Different Preconditioners

Before investigating different preconditioners it is important to point out two characteristic features of $\boldsymbol{\mathcal{S}}_\Lambda$. On one hand, one faces the problem of a scale wise increase of the largest eigenvalue of $\boldsymbol{\mathcal{S}}_\Lambda$, i.e. $\lim \sigma_{\max}(\Lambda) \to \infty$ as $\#\Lambda \to \infty$, which is a consequence of the involved differential operator. On the other hand, it may happen that the smallest eigenvalue $\sigma_{\min}(\Lambda)$ of $\boldsymbol{\mathcal{S}}_\Lambda$ is very close to 0 for all Λ. This is mainly caused by system inherent features, like possibly low observability measure and high stiffness of the differential state equations. Here, the choice of regularization plays an important role and the matrices $\boldsymbol{\mathcal{S}}_\Lambda$ are often very ill–conditioned. This is indeed the case for both test examples. Nevertheless, while we increase the smallest eigenvalue $\sigma_{\min}(\Lambda)$ by varying ε (see Subsection 4.5), the maximal eigenvalue $\sigma_{\max}(\Lambda)$ increasing with the scale of refinement can be bounded independently of Λ by wavelet based preconditioners (see Part 1). More specifically, symmetric, scale depending diagonal preconditioning $\boldsymbol{M}^{1/2}$ with entries of order 2^{-j} for scale j provide condition numbers bounded independently of Λ. Hence, in the following we compare various choices of diagonal entries with this feature.

The simplest preconditioner is the *diagonal matrix*

$$(\boldsymbol{M}^w_\Lambda)_{i,i'} = 2^{-2j}\delta_{i,i'} \tag{138}$$

with $i \hat{=} (l,j,k)$ and $l = 1, \ldots, n_x$ and $(j,k) \in \Lambda$.

Note that the above preconditioning corresponds to a properly scaled wavelet basis in the Sobolev space H^1. One expects that the involved Riesz constants can be improved by normalizing the wavelet basis in the relevant energy space defined by the Schur complement. This suggests to employ as a diagonal preconditioner the

Jacobi preconditioner, namely the matrix $\mathcal{M}_\Lambda^J$ obtained by taking the inverses of the diagonal entries of the Schur complement

$$\left(\boldsymbol{M}_\Lambda^J\right)_{i,i'} := \left((\mathcal{S}_\Lambda)_{i,i}\right)^{-1} \delta_{i,i'} \ . \tag{139}$$

Recall that we wish to avoid expensive matrix multiplications. Therefore, we consider the additional option of *approximating* the diagonal entries of the Schur complement while avoiding its explicit computation. This can be accomplished by exploiting that the involved inverses have, on account of the chosen wavelet discretization, their maximal entries appearing on the diagonal. Therefore, we approximate the diagonal of $\boldsymbol{\mathcal{S}}_\Lambda$ by using their diagonals only. This yields the *approximate Jacobi* preconditioner $\boldsymbol{M}_\Lambda^{aJ}$ with

$$\begin{aligned}\left(\boldsymbol{M}_\Lambda^{aJ}\right)_{i,i'} \quad = \quad & \left(\sum_s \left(\boldsymbol{Q}_\Lambda^{-1} + \boldsymbol{V}_\Lambda \boldsymbol{R}_{\boldsymbol{v}_\Lambda}{}^{-1} \boldsymbol{V}_\Lambda^T\right)^{-1}_{(s,s)} \boldsymbol{C}^2_{\Lambda,(s,i)} + \right. \\ & \left. \sum_s \left(\boldsymbol{W}_\Lambda \boldsymbol{R}_{\boldsymbol{w}_\Lambda}{}^{-1} \boldsymbol{W}_\Lambda^T\right)^{-1}_{(s,s)} \boldsymbol{X}^2_{\Lambda,(s,i)}\right)^{-1} \delta_{i,i'} \ . \end{aligned} \tag{140}$$

Only 4 very sparse vector multiplications are necessary for each entry to determine $\boldsymbol{M}_\Lambda^{aJ}$ (Part 1).

Other scaling strategies with the same qualitative behavior are conceivable. In order to see to which extent the exact values of the scaling weights matter we have applied analogous diagonal preconditioners of the following form. Denoting by $\boldsymbol{\mathcal{S}}_\Lambda^{(i)}$ the i–th row of $\boldsymbol{\mathcal{S}}_\Lambda$ we set

$$\left(\boldsymbol{M}_\Lambda^{\ell_1}\right)_{i,i'} :=\parallel \boldsymbol{\mathcal{S}}_\Lambda^{(i)} \parallel_{\ell_1}^{-1} \delta_{i,i'}, \quad \left(\boldsymbol{M}_\Lambda^{\ell_2}\right)_{i,i'} :=\parallel \boldsymbol{\mathcal{S}}_\Lambda^{(i)} \parallel_{\ell_2}^{-1} \delta_{i,i'} \ . \tag{141}$$

Their disadvantage is that $\boldsymbol{\mathcal{S}}_\Lambda$ has to be assembled.

The results recorded in the following tables confirm the above statements. Table 9 lists the condition numbers and Table 10 gives the iteration numbers for *Example 2.1.* Exact and approximate Jacobi preconditioning $\boldsymbol{M}_\Lambda^J, \boldsymbol{M}_\Lambda^{aJ}$ yield exactly the same results for condition and iteration numbers and are therefore not listed separately. This reveals that exact determination of the diagonal of the Schur complement is not necessary. Obviously, the condition numbers of $\boldsymbol{\mathcal{S}}_\Lambda$ grow with increasing scale and the iteration numbers roughly double for each mesh bisection. As predicted the preconditioners keep the condition numbers bounded but make them settle on a very high level. Moreover, as expected by the condition numbers for lower scales $\boldsymbol{M}_\Lambda^w$ does not reduce the number of iterations very much. $\boldsymbol{M}_\Lambda^{aJ}$ (as well as $\boldsymbol{M}_\Lambda^J$) lead to a nearly constant iteration number for higher scales. The approximate Jacobi preconditioner $\boldsymbol{M}_\Lambda^{aJ}$ appears to offer the by far best compromise between computational cost and efficiency.

The condition number would predict much higher iteration numbers. However, here the superconvergent behavior of the cg–method plays an important role [9, 134].

Table 9: cond $\left(\boldsymbol{M}_\Lambda^{1/2}\boldsymbol{\mathcal{S}}_\Lambda\boldsymbol{M}_\Lambda^{1/2}\right)$ for various preconditioners, *Example 2.1.*

Scale	$\boldsymbol{I}$	$\boldsymbol{M}_\Lambda^{\ell_1}$	$\boldsymbol{M}_\Lambda^{\ell_2}$	$\boldsymbol{M}_\Lambda^{aJ}$	$\boldsymbol{M}_\Lambda^{w}$
2	7.5222e+06	2.5464e+06	2.3050e+06	1.8998e+06	2.7637e+07
3	6.8613e+06	4.2752e+06	3.9818e+06	4.5289e+06	5.1785e+07
4	2.1416e+07	9.0804e+06	7.5754e+06	8.7587e+06	7.8658e+07
5	8.2050e+07	1.6244e+07	1.1043e+07	1.1880e+07	9.4910e+07
6	3.1474e+08	2.6204e+07	1.4261e+07	1.3354e+07	1.0040e+08
7	1.2185e+09	3.9772e+07	1.8236e+07	1.4006e+07	1.0194e+08

Table 10: Iteration numbers for various preconditioners, *Example 2.1.*

Scale	$\boldsymbol{I}$	$\boldsymbol{M}_\Lambda^{\ell_1}$	$\boldsymbol{M}_\Lambda^{\ell_2}$	$\boldsymbol{M}_\Lambda^{aJ}$	$\boldsymbol{M}_\Lambda^{w}$
2	28	22	22	23	30
3	48	39	39	41	56
4	84	66	54	53	77
5	143	91	63	58	84
6	239	112	74	61	89
7	411	134	86	62	90

Figure 26 displays the distribution of eigenvalues for the Schur complement on scale 7. It confirms the expected scale wise increase of the eigenvalues caused by the derivative in the constraints. It also shows that this increase is prevented by the approximate Jacobi preconditioner. As predicted the preconditioner $\boldsymbol{M}_\Lambda^{aJ}$ only affects the eigenvalues for high frequency components. Figure 26 also confirms that

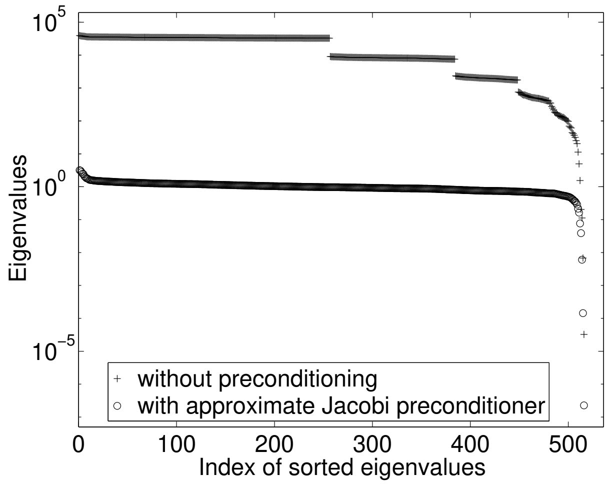


Figure 26: Eigenvalues of $\boldsymbol{\mathcal{S}}_\Lambda$ for the discretization of *Example 2.1.*

the smallest eigenvalue is always close to zero reflecting poor observability of the problem, the effect of the discretization as well as the insufficient amount of regularization. Unfortunately, the smallest eigenvalue is even decreased further by the preconditioner from roughly 10^{-5} to 10^{-7}. Moreover, apart from a few very small eigenvalues the eigenvalues concentrate around one value. For the simplified Uzawa scheme this leads to the superconvergent convergence behavior. While the error reduction in the first iteration steps may be small, subsequently we have a high convergence order since the iterates are close to a subspace which are well separated from the eigenspace of small eigenvalues.

The condition numbers of *Example 2.2* do not quite exhibit the same scale dependent increase up to the computed scale 7 (see Table 11). The effect of the differential operator is in this case strongly dominated by the system inherent properties. Thus, up to scale 7 $\boldsymbol{M}_\Lambda^w$ does not have the desired effect, while the other preconditioners incorporate $\boldsymbol{\mathcal{S}}_\Lambda$ and therefore lead to a reduction of the condition numbers. Again, we have not seen any difference between using the exact or the approximated Schur complement ($\boldsymbol{M}_\Lambda^J$ or $\boldsymbol{M}_\Lambda^{aJ}$) for the Jacobi preconditioner, so the results are not listed separately. The actual iteration numbers are much less than indicated by the

Table 11: cond $\left(\boldsymbol{M}_\Lambda^{1/2}\boldsymbol{\mathcal{S}}_\Lambda\boldsymbol{M}_\Lambda^{1/2}\right)$ for various preconditioners, *Example 2.2*.

Scale	$\boldsymbol{I}$	$\boldsymbol{M}_\Lambda^{\ell_1}$	$\boldsymbol{M}_\Lambda^{\ell_2}$	$\boldsymbol{M}_\Lambda^{aJ}$	$\boldsymbol{M}_\Lambda^{w}$
1	6.6811e+08	2.0136e+08	2.2661e+08	3.3955e+08	6.6811e+08
2	1.8370e+08	7.3924e+06	7.4484e+06	1.1337e+07	2.0925e+08
3	1.8198e+08	3.2145e+06	3.5570e+06	3.9910e+06	3.1657e+08
4	1.8383e+08	2.5785e+06	3.7069e+06	1.2942e+06	6.2646e+08
5	1.8631e+08	2.6227e+06	3.8166e+06	8.1949e+05	1.2605e+09
6	1.8781e+08	2.8957e+06	3.7576e+06	1.3404e+06	1.9685e+09
7	2.1555e+08	3.1816e+06	3.7365e+06	2.2054e+06	2.5265e+09

Table 12: Iteration numbers for various preconditioners, *Example 2.2*.

Scale	$\boldsymbol{I}$	$\boldsymbol{M}_\Lambda^{\ell_1}$	$\boldsymbol{M}_\Lambda^{\ell_2}$	$\boldsymbol{M}_\Lambda^{aJ}$	$\boldsymbol{M}_\Lambda^{w}$
1	148	122	118	109	148
2	277	202	207	191	305
3	527	357	355	293	639
4	880	490	473	393	948
5	1111	600	560	456	1124
6	1297	695	582	470	1158
7	1555	704	601	429	918

condition numbers (see Table 12). This feature we attribute again to the superconvergence behavior of the underlying cg–method. Nevertheless, the superconvergent behavior is not as significant as in *Example 2.1* due to the system inherent difficulties which are reflected in the eigenvalue distributions. Starting at scale 4 the iteration numbers of the preconditioned systems begin to settle at a much lower number than without preconditioning.

In summary, among the tested diagonal preconditioners the approximate Jacobi preconditioner performs best although the results are still unsatisfactory. Essentially two further directions suggest themselves as a remedy.

First, one can resort to more sophisticated preconditioners which require of course also more computational effort. First attempts of employing various incomplete versions of Cholesky factorization to the Schur complement have turned out to fail. For example, the classical incomplete Cholesky factorization based on the sparsity pattern of $\boldsymbol{\mathcal{S}}_\Lambda$ breaks down because positivity of the diagonal entries is lost during the process. The same happens with incomplete block–Cholesky factorization, which means that an exact Cholesky factorization is applied to the block structure, while the inversion of the resulting blocks is 'incomplete'. Exact Cholesky factorization applied to an approximated $\boldsymbol{\mathcal{S}}_\Lambda$, where only the diagonals of the involved wavelet matrices are employed, either also breaks down because of a possible loss of positive definiteness of the approximate Schur complement or it is not efficient. Another attempt to have a better grip on the smallest eigenvalues has been to use a Cholesky factorization of $\boldsymbol{\mathcal{S}}_{\Lambda_{j_0}}$ on a lower scale j_0 extended by a scale depending diagonal preconditioning on the higher scales for preconditioning $\boldsymbol{\mathcal{S}}_{\Lambda_J}$ on a higher scale J. This turned out to produce even worse results than applying the approximate Jacobi preconditioner $\boldsymbol{M}_\Lambda^{aJ}$. Stabilizing incomplete LU–factorization by pivoting is ruled out because one would loose symmetry which is essential for cg based schemes such as the Uzawa algorithm. There remains the class of preconditioners involving parameters like the incomplete Cholesky factorization based on a given threshold parameter, or also, though not leading to a symmetric preconditioner, SPAI. The problem is then to choose the parameters so as to obtain a good preconditioner while keeping the cost of determining and applying the preconditioner acceptable. In principle, one can choose the parameters in such a way that the inverses are determined as preconditioners. In that case we are back to applying a direct solver. Further less established preconditioners that primarily affect the smallest eigenvalues are currently under investigation.

The second direction to improve the condition number is to modify the operator $\boldsymbol{\mathcal{S}}$ by regularization, i.e. in our case by varying ε.

4.5 Varying the Regularization Parameter

The key issue for choosing an appropriate ε is, of course, to attain satisfactory estimation quality. This requires establishing a good compromise between stabilizing the problem and minimizing the resulting bias of the estimate. To avoid bias to a best possible extent, the effect of the regularization of $\boldsymbol{w}$ should be kept at a minimum. Hence, for regularization purposes ε should be rather small. Typically, those states directly entering the output equations (131) are estimated with little bias for small ε due to the high weight of the output prediction errors whereas there are significantly larger estimation errors in the other states. If there are no requirements on the relative estimation error quality of the states dictated by the specific estimation problem, it is reasonable to choose ε so as to maximize the smallest eigenvalue of the Schur complement $\boldsymbol{\mathcal{S}}_\Lambda$ to reduce the condition number. This results in a smaller norm of the inverse operator given by the original optimization problem and leads to the best parameter under these circumstances. The quantitative effect of this strategy is recorded in Tables 13–17.

Table 13: Regularization of *Example 2.1* without preconditioning, condition numbers.

Scale\\ε	100	1	10^{-2}	$5 \cdot 10^{-4}$	10^{-6}	10^{-8}
2	7.430e+08	7.522e+06	1.476e+05	1.349e+05	5.873e+07	5.872e+09
3	6.800e+08	6.861e+06	1.091e+05	5.690e+04	2.100e+07	2.098e+09
4	2.124e+09	2.141e+07	3.251e+05	7.491e+04	1.455e+07	1.451e+09
5	8.136e+09	8.205e+07	1.253e+06	2.200e+05	1.316e+07	1.309e+09
6	3.120e+10	3.147e+08	4.841e+06	8.641e+05	1.292e+07	1.275e+09
7	1.208e+11	1.218e+09	1.878e+07	3.381e+06	1.302e+07	1.267e+09

Table 14: Regularization of *Example 2.1* with approximate Jacobi preconditioning, condition numbers.

Scale\\ε	100	1	10^{-2}	$5 \cdot 10^{-4}$	10^{-6}	10^{-8}
2	1.903e+08	1.899e+06	3.216e+04	1.225e+04	1.096e+04	1.096e+04
3	4.540e+08	4.528e+06	6.349e+04	1.434e+04	1.079e+04	1.078e+04
4	8.774e+08	8.758e+06	1.235e+05	2.141e+04	1.204e+04	1.202e+04
5	1.184e+09	1.188e+07	1.733e+05	2.923e+04	1.272e+04	1.265e+04
6	1.326e+09	1.335e+07	1.982e+05	3.429e+04	1.336e+04	1.318e+04
7	1.389e+09	1.400e+07	2.095e+05	3.679e+04	1.403e+04	1.366e+04

Comparing Table 13 and Table 14, we see how the different effects of regularization and preconditioning interact. On one hand, for $\varepsilon \to 0$ the condition numbers of the Schur complement first decrease and then increase again (Table 13). The Schur complement $\boldsymbol{\mathcal{S}}_\Lambda$ defined in (137) tends to the first summand for $\varepsilon \to 0$ and is in general singular in the limit, since the rank of $\boldsymbol{C}$ is typically smaller than the number of columns. On the other hand, assuming that $\boldsymbol{C}$ only selects several states as outputs (as in *Example 2.1*), decreasing ε enhances the diagonal entries of $\boldsymbol{\mathcal{S}}_\Lambda$ contributed by the diagonal entries of the first term $\boldsymbol{C}_\Lambda^T \left(\boldsymbol{I}_{n_y} \otimes \langle\langle \boldsymbol{\Psi}_{\Lambda_y}, \boldsymbol{\Psi}_{\Lambda_y} \rangle\rangle\right) \boldsymbol{C}_\Lambda$ in (137). Thus, Jacobi preconditioning becomes the more efficient the more ε is decreased. The fact that, while the Schur complement becomes more and more ill–conditioned when $\varepsilon \to 0$, the preconditioner works better and better, is confirmed by the results recorded in Table 14. Without preconditioning the best result for *Example 2.1* is obtained for $\varepsilon = 5 \cdot 10^{-4}$ (Table 13), which offers a compromise for small condition numbers while keeping the information of the system to a satisfactory extent.

Table 15 records the iteration numbers obtained when varying ε. The last column lists the corresponding dimension of the solved system. We apply again the *simplified Uzawa* algorithm with approximate Jacobi preconditioning $\boldsymbol{M}_\Lambda^{aJ}$. The overlapping effect of regularization and preconditioning is reflected by the respective iteration numbers. While the original system becomes singular with $\varepsilon \to 0$, with preconditioning less iterations are required. The above suggested choice of $\varepsilon = 5 \cdot 10^{-4}$ together with the yet most promising approximate Jacobi preconditioner reduces the initial number 201 of iterations for scale 7 (Table 4) to 30. Moreover, the latter iteration number seems to stay constant when progressing to higher refinement scales.

Table 15: Regularization of *Example 2.1* with approximate Jacobi preconditioning, iteration numbers.

Scale\\ε	100	1	10^{-2}	$5 \cdot 10^{-4}$	$5 \cdot 10^{-6}$	dim
2	37	23	20	20	20	41
3	58	41	35	28	29	77
4	73	53	37	31	31	149
5	82	58	38	30	30	293
6	88	61	38	30	29	581
7	90	62	39	30	28	1157

Tables 16 and 17 list the condition and iteration numbers corresponding to *Example 2.2*. We normalized the states and outputs and applied the preconditioner $\boldsymbol{M}_\Lambda^{aJ}$. Here regularization and Jacobi preconditioning do not have the same interfering effect as in *Example 2.1*. This is explained by the model matrix $\boldsymbol{C}$ which in this case arises from model reduction and therefore determines the outputs by combinations of several states. Therefore its structure does not lead to a diagonally dominant behavior of $\boldsymbol{\mathcal{S}}_\Lambda$ for small ε. The best choice of ε with respect of the condition numbers has been found in numerical experiments to be roughly 10^{-4}, while $\varepsilon = 10^{-3}$ turns out to provide the lowest number of iterations.

So far, for *Example 2.2* only the *simplified Uzawa* algorithm works reasonably well (Table 7). The combination of normalization, approximate Jacobi preconditioning and simple regularization reduces significantly the condition numbers from initially 10^{20} to 10^{6}. For this extremely ill–conditioned problem, neither iterative nor direct methods could have been reliably applied without preconditioning. Moreover, the superconvergence of the underlying cg–method also leads for this example to much lower iteration numbers than predicted by the condition numbers. The iteration numbers stay roughly around 350 independently of the refinement scale (see Table 17).

Table 16: Regularization of *Example 2.2*, condition numbers.

Scale\ ε	100	1	10^{-3}	10^{-4}	10^{-6}
1	1.5621e+13	1.5696e+11	5.8125e+08	3.3955e+08	4.4458e+08
2	4.7847e+12	4.7897e+10	5.3420e+07	1.1337e+07	7.3007e+07
3	2.3332e+12	2.3347e+10	2.6321e+07	3.9910e+06	7.7545e+07
4	8.6843e+11	8.6874e+09	8.6592e+06	1.2942e+06	8.4559e+07
5	4.9074e+11	4.9106e+09	4.7343e+06	8.1949e+05	8.7582e+07
6	1.0938e+12	1.0946e+10	1.0576e+07	1.3404e+06	8.8499e+07
7	1.9191e+12	1.9201e+10	1.8251e+07	2.2054e+06	8.8698e+07

Table 17: Regularization of *Example 2.2*, iteration numbers.

Scale\ ε	100	1	10^{-3}	10^{-4}	10^{-6}	dim
1	max	185	98	109	174	75
2	max	394	149	191	384	133
3	max	565	224	293	925	249
4	1497	765	306	393	1779	481
5	1606	871	349	456	2507	945
6	1721	911	365	470	2903	1873
7	1753	970	329	429	max	3729

5 Conclusions

In summary, for the wavelet discretization of the optimization problem (129)–(131), the most efficient method tested here is the *simplified Uzawa* algorithm with *approximate Jacobi* preconditioner $\boldsymbol{M}_\Lambda^{aJ}$ applied to the system red$_2$ in combination with normalization of the states and outputs and with an appropriate choice of the parameter ε. The cost of one iteration of the simplified Uzawa algorithm corresponds to one iteration of the cg–method. Moreover, the number of iterations is less than for the application of various other, more expensive Krylov–space methods like *symmlq, bicg* and *gmres*. The combination of normalization, simple regularization and approximate Jacobi preconditioning gives rise to significantly reduced condition numbers. More precisely, the approximate Jacobi preconditioner eliminates the scale wise increase of the relevant eigenvalues and bounds the condition number independently of the employed index set Λ. On the other hand, normalization and simple regularization increases the smallest eigenvalues. Depending on the model matrix $\boldsymbol{C}$ we can observe an interfering effect of preconditioning and regularization with respect to the condition numbers. The choice of the regularization parameter ε should therefore not be guided by the preconditioned systems. In above cases ε ranging between 10^{-3} and 10^{-4} turned out to be suitable. Nevertheless, depending on model inherent features the condition numbers may still be too large. More sophisticated methods have to be developed to raise the smallest eigenvalues. Currently, a Schur complement method is under investigation. In spite of the possibly still large condition numbers the superconvergent behavior of the underlying cg–method yields considerably smaller numbers of iterations. This stems from a clustering of eigenvalues, which is caused by the approximated Jacobi preconditioning. Last but not least the application of an iterative solver does not require assembling the discretization matrices. Only their application to a vector has to be carried out by exploiting the Kronecker product rules.

State Estimation without regularizing the initial data

Luise Blank

Abstract

Monitoring dynamical processes requires the estimation of the entire state, which is only partly accessible by measurements. Most quantities must be determined via model based state estimation, which in general is an ill–posed inverse problem. Regularization techniques have to be applied. To avoid undesired bias we omit the commonly used approach of regularizing the unknown initial data. To the authors knowledge the resulting minimization problem has not been analysed mathematically yet, which is the purpose of this paper. The first order necessary conditions are presented and the optimization problem is reduced by several variables. Hence, e.g. one otherwise necessary regularization parameter is dispensable. The influence of the regularization parameters and of the model matrices is studied in detail for linear models. It is shown that for any choice of regularization parameters the condition numbers of the evolving operators can be arbitrarily large, if the spectral radius of the system matrix is large. In case of one state only we derive additionally bounds for the perturbation of the initial data resulting from measurement errors.

Key Words: state estimation, optimal control, optimality condition, inverse problems, dynamical systems.

1 Introduction

State estimation is the task of determining from a few given, typically noisy measurements over a past time horizon all state functions, which are involved e.g. into a chemical process. The measurements correspond only to a few states. The others have to be estimated using the coupling of the states via model equations. State estimation can be viewed in general an ill–posed inverse problem. Then, regularization techniques have to be applied. In most of the literature the initial data are also regularized (see e.g.[6, 143, 144, 156]). Then the resulting optimization formulation coincides widely with a corresponding control problem, where the initial data are given. Control problems are analysed by many authors and numerical methods are available. However, in state estimation the initial data are typically unknown. Sometimes experiences provide a rough guess for the initial data, or state estimation on one horizon provides guesses for a forwardly moved time horizon, which are then used as reference values for regularization. Though precise values are not available. Hence, to avoid bias we omit the regularization of the initial data (for further discussion see [20]). To the authors knowledge there is still a lack of theoretical investigations to this problem formulation. Thus the purpose of this paper is to analyse the effects of this problem formulation, to gain insight into the difficulties and to offer possibilities to overcome, respectively to reduce them in an analytical or numerical way.

In Section 2 we shortly deduce the problem formulation of state estimation with incorporated model error functions, discuss the ill–posedness and introduce the applied regularization.

Section 3 is concerned with the reduction of the resulting minimization problem. Based on the first oder necessary optimality conditions it turns out that the optimization problem can be formulated without the output functions, the corresponding model error functions and without output equation. In the equations for the optimality conditions also the model error function corresponding to the state equation can be eliminated. In case of linear ODE's as state equations where model error functions are present in all equations and omitting inequality constraints, the necessary conditions reduce to a differential equation of second order for the states with mixed boundary constraints. This differential equation and its corresponding weak formulation are analysed in detail in the next section.

To gain insight into the properties of the relevant operators we study in Section 4, as a start, the linear case with time independent coefficients. Of particular interest is the influence of the regularization parameters and the influence of the coefficients of the model equations.

In Section 4.1 we consider the case of one state only. Assuming fixed regularization parameters and coefficients we show that the regularized inverse problem is well posed with respect to L_2–perturbations and give bounds for the corresponding map. However, large negative system coefficients may lead to large errors in the initial data for small disturbances in the measurements. From the engineering side the more interesting case are disturbances in the L_∞–norm. These lead to a bound only dependent on the regularization parameters. Not only for the numerical approach it is essential to have bounds also on the norm and the condition number of the operator corresponding to the resulting (differential) system. We give upper and lower bounds for the norm and see that they grow to infinity for any regularization parameter, if the coefficient in the model equation goes to infinity. Furthermore, the connection to the Sturm–Liouville's eigenvalue problem yields information about the eigenvalues. An exponential eigenfunction corresponds to the smallest eigenvalue. This result serves in [32] as one essential ingredient to obtain an efficient preconditioner for the numerical solution of the resulting linear problem.

In Section 4.2 we extend the results concerning the bounds of the operator to a system with several states. Here the issue of observability comes into play. Not only a large spectral radius of the system matrix can lead to large condition numbers but also eigenvectors of the system matrix which are close to the null space of the matrix, which determines the output functions from the states.

Finally in Section 5 we summarize the conclusions.

2 Derivation of the Problem Formulation

In [20] the mathematical formulation of the estimation problem is discussed and its engineering interpretation is given in detail. In this paper we concentrate on the more mathematical approach, arguing with the mathematical features of the variables and equations. We state the model equations and their extension by model error functions and then discuss shortly the regularization of the problem.

2.1 Model Equations

The quantities of interest in process monitoring can be summarized in the state functions $\boldsymbol{x}$, the output functions $\boldsymbol{y}$, the control variables $\boldsymbol{u}$ and parameters $\boldsymbol{p}$. For the particular case of state estimation the controls $\boldsymbol{u}$ and the parameters $\boldsymbol{p}$ are known variables. Then, the mathematical description of the state equations of the process is assumed to be given by the differential algebraic equations (DAE's)

$$\boldsymbol{G}\dot{\boldsymbol{x}} - \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}) = \boldsymbol{0} \tag{142}$$

with the consistent initial condition [37].

$$\boldsymbol{x}(0) = \boldsymbol{x}_0. \tag{143}$$

The output equations

$$\boldsymbol{y} - \boldsymbol{C}\boldsymbol{x} = \boldsymbol{0} \tag{144}$$

describe which states can be measured. Here $\boldsymbol{C}$ is a time independent matrix, typically an identity matrix extended by zero columns. The number of output functions n_y are typically far less than the number of involved states n_x. We require that the system is observable [50, 53, 97], i.e. the map $\mathcal{F} : \boldsymbol{x}_0 \longmapsto \boldsymbol{y}$ given by (142)–(144) is injective for any given $\boldsymbol{u}$ and $\boldsymbol{p}$. Hence given an output $\boldsymbol{y} \in Im(\mathcal{F})$ we can reconstruct uniquely $\boldsymbol{x}$ using (142) and (144).

To include possible model errors into the model equations the DAE's are extended linearly by error functions $\boldsymbol{w}$ and $\boldsymbol{v}$, which results into the equations

$$\boldsymbol{G}\dot{\boldsymbol{x}} - \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}) - \boldsymbol{W}\boldsymbol{w} = \boldsymbol{0} \tag{145}$$

$$\boldsymbol{y} - \boldsymbol{C}\boldsymbol{x} - \boldsymbol{V}\boldsymbol{v} = \boldsymbol{0}. \tag{146}$$

The matrices $\boldsymbol{W}$ and $\boldsymbol{V}$ are typically identity matrices enlarged by zero rows. They reflect which equations are trusted. The above model equations (145)–(146) and (143) describe the process on a given interval (time horizon on $[0, T]$) which is assumed to be projected onto $[0, 1]$. For convenience we omit the interval $[0, 1]$ in the following. Moreover, we only assume that the equations are fulfilled in the weak sense. Consequently, the corresponding function spaces are L_2 and the Sobolev space

H^1. For simplicity if the dimension $n = n_x$ etc. is obvious in the context, we omit the exponent n and write L_2 and H^1 instead of the precise notation $(H^1)^n$.

Inequality constraints reflect, for example, safety restrictions and control limits and can be written as

$$\boldsymbol{c}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}) \leq \mathbf{0}. \tag{147}$$

For state estimation they are typically of the form $\boldsymbol{c}_i(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}) = x_i - \bar{c}_i$ and $\boldsymbol{c}_{i+n_x}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}) = -x_i + \bar{c}_{i+n_x}$ for $i = 1, \ldots, n_x$.

2.2 The Inverse Problem and Regularization

Given $\boldsymbol{x}_0, \boldsymbol{w}$ and $\boldsymbol{v}$ the solution $\boldsymbol{y}$ is uniquely determined by (145) and (146). This defines an operator (in case of $\boldsymbol{G} = \boldsymbol{I}$)

$$\boldsymbol{T}\begin{pmatrix} \boldsymbol{x}_0 \\ \boldsymbol{w} \\ \boldsymbol{v} \end{pmatrix} := \boldsymbol{y} \quad \text{with} \quad \boldsymbol{T} : R^{n_x} \times (L_2)^{n_w} \times \left(H^1\right)^{n_v} \longrightarrow \left(H^1\right)^{n_y}. \tag{148}$$

For state estimation one wants to determine $\boldsymbol{x}_0, \boldsymbol{w}$ and $\boldsymbol{v}$ and, hence, $\boldsymbol{x}$ from given data. If the data are clean they shall with $\boldsymbol{y}$. However, in general only noisy measurements are available. Assuming that the measurements taken at discrete times can be processed to signal functions $\boldsymbol{z}$, these functions are typically not in H^1 but in L_2. Hence, the output functions $\boldsymbol{y}$ cannot match the measurements $\boldsymbol{z}$ exactly. The estimation solution is given as a least squares solution, i.e. $\boldsymbol{T}^{\dagger}\boldsymbol{z} := \left(\boldsymbol{x}_0^T, \boldsymbol{w}^T, \boldsymbol{v}^T\right)^T$ where $\|\boldsymbol{T}\left(\boldsymbol{x}_0^T, \boldsymbol{w}^T, \boldsymbol{v}^T\right)^T - \boldsymbol{z}\|_{(L_2)^{n_y}}$ is minimal. Taking into account that each output function can be weighted separately the optimization problem is formulated as

$$\min_{\boldsymbol{x}_0, \boldsymbol{w}, \boldsymbol{v}} 1/2 \int_0^1 (\boldsymbol{y} - \boldsymbol{z})^T \boldsymbol{R}_y (\boldsymbol{y} - \boldsymbol{z})\, dt \qquad \text{with} \qquad \boldsymbol{T}\left(\boldsymbol{x}_0^T, \boldsymbol{w}^T, \boldsymbol{v}^T\right) = \boldsymbol{y} \tag{149}$$

with some time independent matrix $\boldsymbol{R}_y$.

The generalized inverse $\boldsymbol{T}^{\dagger} : (L_2)^{n_y} \longrightarrow R^{n_x} \times (L_2)^{n_w} \times (H^1)^{n_v}$ is an unbounded operator, since highly oscillatory functions $\boldsymbol{z} \in (H^1)^{n_y}$ with uniformly bounded L_2– or L_∞–norms can lead to unbounded L_2–, respectively, L_∞–norms of $\boldsymbol{w}$ due to the involved differential operator (for example $z(t) = \varepsilon \sin(t/\varepsilon^2)$). Additionally the problem of non–uniqueness in $\boldsymbol{w}$ and $\boldsymbol{v}$ can occur.

One possible regularization technique to overcome the ill–posedness is Tikhonov–type regularization [77, 117]. This leads to the following ansatz for the cost functional:

$$1/2 \int_0^1 (\boldsymbol{y} - \boldsymbol{z})^T \boldsymbol{R}_y (\boldsymbol{y} - \boldsymbol{z}) + \boldsymbol{v}^T \boldsymbol{R}_v \boldsymbol{v} + \boldsymbol{w}^T \boldsymbol{R}_w \boldsymbol{w}\, dt \tag{150}$$

The resulting optimization problem for state estimation on a fixed horizon projected onto $[0, 1]$ is then given by minimizing (150) with the model equations (145)–(146) and the initial condition (143) as equality constraints and the inequality constraints

(147). We would like to mention that typically the initial values $\boldsymbol{x}_0$ are regularized too (for e.g. [108, 144, 157]). However, we omit this on purpose, as mentioned in the beginning, to avoid unnecessary bias.

3 Reduction of the Problem Formulation

Since $\boldsymbol{x}_0$ is unknown we observe that the initial condition (143) has in case of an estimation problem no information and need for the optimization problem. Hence, (143) can be omitted in the problem formulation. Moreover, we do not have to distinguish between $\boldsymbol{u}$ and $\boldsymbol{p}$ for analytical reasons, since both are given. In the following we include $\boldsymbol{p}$ into $\boldsymbol{u}$.

In the next step we exploit the relations to the Lagrange multipliers corresponding to the equality constraints. Therefore, we consider in detail the first order necessary optimality conditions leading to the Euler–Lagrange equations, which are obtained by the Pontryagin's maximum principle [95, 98, 152].

Since no inequality constraints on $\boldsymbol{y}$ and $\boldsymbol{v}$ are present, we obtain —using the Lagrange multiplier $\boldsymbol{\lambda}_2$ with respect to (146)— the equations

$$\begin{aligned} \boldsymbol{y} &= -{\boldsymbol{R}_y}^{-1}\boldsymbol{\lambda}_2 + \boldsymbol{z} && (151)\\ \boldsymbol{v} &= {\boldsymbol{R}_v}^{-1}\boldsymbol{V}^T\boldsymbol{\lambda}_2 && (152) \end{aligned}$$

and can deduce

$$\boldsymbol{v} = {\boldsymbol{R}_v}^{-1}\boldsymbol{V}^T\boldsymbol{R}_y(\boldsymbol{z}-\boldsymbol{y}). \tag{153}$$

Using (146) and (153) yields

$$\boldsymbol{y} = \left(\boldsymbol{I} + \boldsymbol{V}{\boldsymbol{R}_v}^{-1}\boldsymbol{V}^T\boldsymbol{R}_y\right)^{-1}(\boldsymbol{C}\boldsymbol{x}-\boldsymbol{z}) + \boldsymbol{z}. \tag{154}$$

Consequently $\boldsymbol{y}$ and $\boldsymbol{v}$ can be eliminated in the formulation of the optimization problem. Let us define now $\boldsymbol{Q} := \left({\boldsymbol{R}_y}^{-1} + \boldsymbol{V}{\boldsymbol{R}_v}^{-1}\boldsymbol{V}^T\right)^{-1}$ then the problem formulation (145)–(147), (150) is equivalent to the following minimization.

Reduced optimization problem:

$$\min 1/2 \int_0^1 (\boldsymbol{C}\boldsymbol{x}-\boldsymbol{z})^T\boldsymbol{Q}(\boldsymbol{C}\boldsymbol{x}-\boldsymbol{z}) + \boldsymbol{w}^T\boldsymbol{R}_w\boldsymbol{w}\; dt \tag{155}$$

s.t.

$$\begin{aligned} \boldsymbol{G}\dot{\boldsymbol{x}} - \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u}) - \boldsymbol{W}\boldsymbol{w} &= \boldsymbol{0} && (156)\\ \boldsymbol{c}(\boldsymbol{x},\boldsymbol{u}) &\le \boldsymbol{0}. && (157) \end{aligned}$$

Considering the obtained optimization problem we can deduce:

Remark 3.1 *The inclusion of $\boldsymbol{v}$ as a model error function into the problem formulation is not necessary. We can view $\boldsymbol{Cx}$ as in (144) as outputs and, after obtaining the estimates, distinguish as in (146) which amount $\boldsymbol{y}$ of the outputs $\boldsymbol{Cx}$ is trustful and which amount is considered as error $\boldsymbol{v}$. Consequently, we can also avoid the problem of finding an appropriate regularization matrix $\boldsymbol{R}_v$.*

Exploiting the Lagrange multiplier $\boldsymbol{\lambda}$ with respect to (156) we obtain the relation

$$\boldsymbol{w} = \boldsymbol{R}_w{}^{-1}\boldsymbol{W}^T\boldsymbol{\lambda}, \tag{158}$$

which means that we can eliminate the model error function $\boldsymbol{w}$ in the optimization process. However, with this elimination we leave also the framework of the saddle point formulation. Given (158) $\boldsymbol{w}$ has the same differentiability as $\boldsymbol{\lambda}$. Defining $\boldsymbol{R} := \boldsymbol{W}\boldsymbol{R}_w{}^{-1}\boldsymbol{W}^T$ we make the following substitutions

$$\begin{aligned} \boldsymbol{Ww} &= \boldsymbol{R\lambda} && (159)\\ \boldsymbol{w}^T\boldsymbol{R}_w\boldsymbol{w} &= \boldsymbol{\lambda}^T\boldsymbol{R\lambda}. && (160) \end{aligned}$$

This clearly indicates that for the computation only $\boldsymbol{R}$ plays a role.

Remark 3.2 *It is not necessary to distinguish between $\boldsymbol{W}$ and $\boldsymbol{R}_w$ until one wants to determine $\boldsymbol{w}$ instead of $\boldsymbol{Ww}$.*

Necessary first order conditions:

The remaining first order necessary optimality conditions for the solution of the estimation problem are the equations

$$\begin{aligned} \boldsymbol{G}\dot{\boldsymbol{x}} - \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u}) - \boldsymbol{R\lambda} &= \boldsymbol{0} && (161)\\ -\frac{d}{dt}\left(\boldsymbol{G}^T\boldsymbol{\lambda}\right) - \left(\frac{\partial}{\partial \boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})\right)^T\boldsymbol{\lambda} + \boldsymbol{C}^T\boldsymbol{QCx} + \left(\frac{\partial}{\partial \boldsymbol{x}}\boldsymbol{c}(\boldsymbol{x},\boldsymbol{u})\right)^T\boldsymbol{\nu} &= \boldsymbol{C}^T\boldsymbol{Qz} && (162)\\ \left(\boldsymbol{G}^T\boldsymbol{\lambda}\right)(0) &= \boldsymbol{0} && (163)\\ \left(\boldsymbol{G}^T\boldsymbol{\lambda}\right)(1) &= \boldsymbol{0} && (164)\\ \boldsymbol{\nu}^T\boldsymbol{c}(\boldsymbol{x},\boldsymbol{u}) &= 0 && (165)\\ \boldsymbol{c}(\boldsymbol{x},\boldsymbol{u}) &\leq \boldsymbol{0} && (166)\\ \boldsymbol{\nu} &\geq \boldsymbol{0} && (167) \end{aligned}$$

with unknown Lagrange multipliers $\boldsymbol{\lambda}$, $\boldsymbol{\nu}$ and unknown states $\boldsymbol{x}$. The output function $\boldsymbol{y}$ is given by (154), the model error functions $\boldsymbol{v}$ by (153) and $\boldsymbol{w}$ by (158). Equations (163) and (164) reflect the fact that no boundary conditions and no regularization of the boundary values of the states $\boldsymbol{x}$ are given.

In case $\boldsymbol{R}_w = \boldsymbol{0}$ which is for example considered in case of regularization by discretization [29], $\boldsymbol{w}$ cannot be eliminated. We obtain additionally the equation $\boldsymbol{W}^T\boldsymbol{\lambda} = \boldsymbol{0}$ and we have to substitute $\boldsymbol{R\lambda}$ with $\boldsymbol{Ww}$ in (161). If no inequality constraints are present, (161)–(164) is a DAE system of order one, where for at most half of the variables boundary conditions are given, while for the others the boundary values are unknown. In case that the state equations are given by ODE's only, we obtain with (158)

Remark 3.3 *If $\boldsymbol{G} = \boldsymbol{I}$ then the model error functions $\boldsymbol{w}$ are in H^1 and vanish at the boundary $t = 0$ and $t = 1$. The boundary values are enforced by the described Tikhonov–regularization.*

It is questionable whether one would like to have possible model errors also at the boundary. Regularization by discretization may offer this possibility. In case of regularization of the initial values of the states $\boldsymbol{x}$ with some reference guesses $\bar{\boldsymbol{x}}_0$, which is seen mostly in the literature, the model errors are free at the boundary too. However, as mentioned, for wrong reference data undesired bias may appear in the states.

The weak formulation can play an important role in the numerical treatment of the optimization problem (e.g. applying Galerkin methods). While the weak formulation of (161) and (165) is obvious, for further references we would like to formulate at this point the weak formulation of (162) which includes the boundary conditions (163) and (164). Denoting with $\langle\, , \rangle$ the standard L_2–product, we obtain:

$$\begin{aligned} \langle \boldsymbol{G}^T\boldsymbol{\lambda}, \frac{d}{dt}\boldsymbol{\zeta}\rangle \; &- \; \langle\left(\frac{\partial}{\partial \boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})\right)^T \boldsymbol{\lambda}, \boldsymbol{\zeta}\rangle + \langle \boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}\boldsymbol{x}, \boldsymbol{\zeta}\rangle + \langle\left(\frac{\partial}{\partial \boldsymbol{x}}\boldsymbol{c}(\boldsymbol{x},\boldsymbol{u})\right)^T \boldsymbol{\nu}, \boldsymbol{\zeta}\rangle \\ &= \; \langle \boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{z}, \boldsymbol{\zeta}\rangle \qquad \text{for all } \boldsymbol{\zeta} \in H^1. \end{aligned} \tag{168}$$

3.1 Invertible Regularization Matrix R

If $\boldsymbol{R}$ is invertible, model error functions have to be present in each state equation. This implies that $\boldsymbol{W}$ is invertible. Hence $\boldsymbol{w}$ can be replaced in the cost functional by $\boldsymbol{W}^{-1}(\boldsymbol{G}\dot{\boldsymbol{x}} - \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u}))$, which leads to an optimization problem in $\boldsymbol{x}$ with the inequality constraints (166) only.

Consequently, also $\boldsymbol{\lambda}$ can be eliminated using equation (161). It follows

$$\boldsymbol{\lambda} \;=\; \boldsymbol{R}^{-1}(\boldsymbol{G}\dot{\boldsymbol{x}} - \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})) \tag{169}$$

and we obtain a second order DAE system for the states $\boldsymbol{x}$ (see (162)):

$$\begin{aligned} &-\frac{d}{dt}\left(\boldsymbol{G}^T\boldsymbol{R}^{-1}\boldsymbol{G}\frac{d}{dt}\boldsymbol{x}\right) + \frac{d}{dt}\left(\boldsymbol{G}^T\boldsymbol{R}^{-1}\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})\right) - \left(\frac{\partial}{\partial \boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})\right)^T \boldsymbol{R}^{-1}\boldsymbol{G}\frac{d}{dt}\boldsymbol{x} \\ &\quad + \left(\frac{\partial}{\partial \boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})\right)^T \boldsymbol{R}^{-1}\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u}) + \boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}\boldsymbol{x} + \left(\frac{\partial}{\partial \boldsymbol{x}}\boldsymbol{c}(\boldsymbol{x},\boldsymbol{u})\right)^T \boldsymbol{\nu} = \boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{z} \end{aligned} \tag{170}$$

with the boundary constraints $\boldsymbol{G}^T\boldsymbol{R}^{-1}(\boldsymbol{G}\dot{\boldsymbol{x}} - \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})) = \boldsymbol{0}$ for $t = 0$ and $t = 1$ and the equations (165)–(167) for the inequality constraints.

In general $\boldsymbol{G}$ and $\boldsymbol{R}$ are time independent matrices, hence the differentiation in time can be interchanged in the above equation with these matrices. Moreover, we would like to remind that $\boldsymbol{R}$ and $\boldsymbol{Q}$ are symmetric and positive definite.

Linear ODE's without inequality constraints

In the linear ODE case, i.e. $\boldsymbol{G}\dot{\boldsymbol{x}} - \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) = \dot{\boldsymbol{x}} - \boldsymbol{A}\boldsymbol{x} - \boldsymbol{u}$ without inequality constraints where we additionally summarized all given parameters, controls and their perhaps nonlinear influence to a given function $\boldsymbol{u}$, the optimization problem is equivalent to the unconstraint minimization problem

$$min\left\{\|\boldsymbol{Q}^{1/2}\left(\boldsymbol{C}\boldsymbol{x} - \boldsymbol{z}\right)\|_{L_2}^2 + \|\boldsymbol{R}^{-1/2}\left(\dot{\boldsymbol{x}} - \boldsymbol{A}\boldsymbol{x} - \boldsymbol{u}\right)\|_{L_2}^2\right\}. \tag{171}$$

As necessary first order conditions we obtain the second order differential equation (see also (170))

$$\begin{aligned} & -\boldsymbol{R}^{-1}\ddot{\boldsymbol{x}} + \left(\boldsymbol{R}^{-1}\boldsymbol{A} - \boldsymbol{A}^T\boldsymbol{R}^{-1}\right)\dot{\boldsymbol{x}} + \left(\boldsymbol{A}^T\boldsymbol{R}^{-1}\boldsymbol{A} + \boldsymbol{R}^{-1}\dot{\boldsymbol{A}} + \boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}\right)\boldsymbol{x} \\ = \;& \boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{z} - \boldsymbol{R}^{-1}\dot{\boldsymbol{u}} - \boldsymbol{A}^T\boldsymbol{R}^{-1}\boldsymbol{u} \end{aligned} \tag{172}$$

with the boundary conditions $\dot{\boldsymbol{x}} - \boldsymbol{A}\boldsymbol{x} = \boldsymbol{u}$ at $t = 0$ and $t = 1$.

While the strong formulation does not look symmetric, the weak formulation reflects the symmetry given by the formulation (171).

$$\begin{aligned} & \langle\dot{\boldsymbol{\zeta}} - \boldsymbol{A}\boldsymbol{\zeta}, \boldsymbol{R}^{-1}\left(\dot{\boldsymbol{x}} - \boldsymbol{A}\boldsymbol{x}\right)\rangle + \langle\boldsymbol{C}\boldsymbol{\zeta}, \boldsymbol{Q}\boldsymbol{C}\boldsymbol{x}\rangle \\ = \;& \langle\boldsymbol{C}\boldsymbol{\zeta}, \boldsymbol{Q}\boldsymbol{z}\rangle + \langle\dot{\boldsymbol{\zeta}} - \boldsymbol{A}\boldsymbol{\zeta}, \boldsymbol{R}^{-1}\boldsymbol{u}\rangle \qquad \text{for all } \boldsymbol{\zeta} \in H^1. \end{aligned} \tag{173}$$

4 Properties of the involved operators

To gain more insight in the difficulties for the solution process we consider the linear case, with time independent coefficients and without inequality constraints in more detail. It enhances already some of the main features we face for state estimation if the initial data are not regularized in the problem formulation. We first consider a model with one state only. In this case we can determine the solution explicitly and analyse its dependency on perturbation in the signal $\boldsymbol{z}$. For the numerical approach however, equation (173) or equivalent formulations are used. Hence, the properties of the operator corresponding to (173) is analysed in more detail. In the second subsection we then can extend most ideas to systems with several states taking into account the issue of observability.

4.1 Linear model with one state function

With $n_x = 1$, $\boldsymbol{A} = \alpha$, $\boldsymbol{C} = \delta$, $\boldsymbol{W} = 1$, $\boldsymbol{R}_w = r$ and $\boldsymbol{Q} = q$ above differential equations reduce to

$$\begin{aligned} -r\ddot{x} + \left(r\alpha^2 + q\delta^2\right)x &= q\delta z - r\dot{u} - r\alpha u \qquad (174) \\ \text{s.t.} \quad \dot{x}(1) - \alpha x(1) = u(1) \quad &\text{and} \quad \dot{x}(0) - \alpha x(0) = u(0). \end{aligned}$$

The solution x and the model error function w are given with $\zeta := \sqrt{\alpha^2 + \frac{q\delta^2}{r}}$ by:

$$x_0 = \frac{-1}{\sinh(\zeta)} \int_0^1 \{\sinh(\zeta - \zeta s)u(s) \tag{175}$$
$$+\frac{1}{\delta}\left[\alpha \sinh(\zeta - \zeta s) - \zeta \cosh(\zeta - \zeta s)\right] z(s)\Big\} \, ds$$

$$x(t) = \frac{1}{\zeta}\left(\zeta \cosh(\zeta t) + \alpha \sinh(\zeta t)\right) x_0 \tag{176}$$
$$+\frac{1}{\zeta}\int_0^t \left\{\left(\zeta \cosh(\zeta(t-s)) + \alpha \sinh\left(\zeta(t-s)\right)\right) u(s) - \frac{q\delta}{r}\sinh\left(\zeta(t-s)\right) z(s)\right\} ds$$

$$w(t) = \frac{q\delta^2}{\zeta r}\sinh(\zeta t) x_0 \tag{177}$$
$$+\frac{q\delta^2}{\zeta r}\int_0^t \left\{\sinh\left(\zeta(t-s)\right) u(s) + \frac{1}{\delta}\left(\alpha \sinh\left(\zeta(t-s)\right) - \zeta \cosh(\zeta(t-s))\right) z(s)\right\} ds.$$

One typically has $\delta = 1$ meaning the state x corresponds to the measurements z in the output equation (146). The parameters influencing the solution are the regularization parameters and the coefficient α of the ODE.

Theorem 4.1 *Assuming a linear model with one state function and time independent coefficients and omitting inequality constraints we obtain with above notation:*

1. *The regularized problem formulation (155)–(157) of state estimation, i.e. given z determining x_0 and w, is well posed.*

2. *Small perturbation of z in the L_2–norm can lead to unbounded error propagation in the initial data x_0 independently of r and q for $\alpha \longrightarrow -\infty$.*

3. *For perturbations of z in the L_∞–norm we have bounds independently of α:*

$$|x_0^\varepsilon| \le \frac{2}{|\delta|}\|z^\varepsilon\|_{L_\infty} \quad \textit{and} \quad |w^\varepsilon(t)| \le \frac{2q\delta}{r\zeta(1-e^{-2\zeta})}\|z^\varepsilon\|_{L_\infty} \le \frac{2\sqrt{q/r}}{1-e^{-2\delta\sqrt{q/r}}}\|z^\varepsilon\|_{L_\infty}$$

where we used the abbreviation $z^\varepsilon = z - \bar{z}$ etc. for the perturbation.

Proof: Let us denote $f(t,s) := \zeta \cosh(\zeta(t-s)) - \alpha \sinh\left(\zeta(t-s)\right)$ then $f > 0$ for all t, s and

$$\|\frac{1}{\sinh(\zeta)} f(1,.)\|_{L_2} = \left[\frac{q\delta^2}{2r\sinh^2(\zeta)} + \frac{\zeta^2+\alpha^2}{2\zeta}\frac{\cosh(\zeta)}{\sinh(\zeta)} - \alpha\right]^{1/2}.$$

Hence $\|\frac{1}{\sinh(\zeta)} f(1,.)\|_{L_2} \approx \sqrt{2|\alpha|}$ as $\alpha \longrightarrow -\infty$.

Moreover we have

$$\|f(.,.)\|^2_{L_2((0,1)\times(0,1))} = \frac{q\delta^2}{2r} + (1 - \frac{q\delta^2}{2(\alpha^2 r + q\delta^2)}) \sinh^2(\zeta).$$

1.) With the Hölder–inequality we have

$$|x_0^\epsilon| \le \|\frac{1}{\delta\sinh(\zeta)} f(1,.)z^\epsilon\|_{L_1} \le \|\frac{1}{\delta\sinh(\zeta)} f(1,.)\|_{L_2}\|z^\epsilon\|_{L_2} \le c\|z^\epsilon\|_{L_2}$$

for fixed r, q, δ and α. Also we obtain the bound

$$\|w^\epsilon\|_{L_2} \le \left(\|\frac{1}{\delta\sinh(\zeta)} f(1,.)\|_{L_2} \|\sinh(\zeta\cdot)\|_{L_2} + \|\frac{1}{\delta} f(.,.)\|_{L_2((0,1)\times(0,1))} \right) \frac{q\delta^2}{r\zeta} \|z^\epsilon\|_{L_2}.$$

2.) Choosing $z^\epsilon(s) = \frac{1}{\sqrt{|\alpha|}} \varepsilon\delta \frac{1}{\sinh(\zeta)} f(1,s)$ we have $\|z^\epsilon\|_{L_2} \le c\varepsilon$ with c independent of α. On the other hand

$$x_0^\varepsilon = \frac{1}{\sqrt{|\alpha|}} \varepsilon \|\frac{1}{\sinh(\zeta)} f(1,.)\|^2_{L_2} \approx 2\varepsilon\sqrt{|\alpha|} \quad \text{as } \alpha \longrightarrow -\infty.$$

3.) For the bound of w^ε we use the following transformation

$$\begin{aligned} w^\varepsilon(t) &= \frac{q\delta}{r\zeta\sinh(\zeta)} \Big\{ \sinh(\zeta t) \int_t^1 f(1,s) z^\varepsilon(s) ds \\ &\quad + \int_0^t \{\sinh(\zeta t) f(1,s) - \sinh(\zeta) f(t,s)\} z^\varepsilon(s) ds \Big\} \\ &= \frac{q\delta}{r\zeta\sinh(\zeta)} \Big\{ \sinh(\zeta t) \int_t^1 f(1,s) z^\varepsilon(s) ds - \sinh(\zeta - \zeta t) \int_0^t f(0,s) z^\varepsilon(s) ds \Big\}, \end{aligned}$$

$$\begin{aligned} |w^\varepsilon(t)| &\le \frac{q\delta}{r\zeta} \|z^\varepsilon\|_{L_\infty} \Big\{ \frac{\sinh(\zeta t)}{\sinh(\zeta)} \Big(\sinh(\zeta - \zeta t) - \frac{\alpha}{\zeta}(\cosh(\zeta - \zeta t) - 1) \Big) \\ &\quad + \frac{\sinh(\zeta - \zeta t)}{\sinh(\zeta)} \Big(\sinh(\zeta t) + \frac{\alpha}{\zeta}(\cosh(\zeta t) - 1) \Big) \Big\} \\ &\le \frac{q\delta}{r\zeta} \|z^\varepsilon\|_{L_\infty} \frac{2e^\zeta}{e^\zeta - e^{-\zeta}}. \end{aligned}$$

Moreover we can conclude

$$|x_0^\varepsilon| \le \|\frac{1}{\delta\sinh(\zeta)} f(1,.)\|_{L_1} \|z^\varepsilon\|_{L_\infty} < \frac{2}{|\delta|} \|z^\varepsilon\|_{L_\infty}.$$

q.e.d.

For the numerical approach not only the map $z \longmapsto (x_0, w)$ is of interest but also the properties of the discretization matrix corresponding to the equation (173), respectively to (174). Let us define the symmetric bilinear form $a : H^1 \times H^1 \to R$

$$a(\xi, x) \;=\; r\langle \dot{\xi} - \alpha\xi, \dot{x} - \alpha x\rangle + q\delta^2\langle \xi, x\rangle. \tag{178}$$

Obviously, a is positive definite if $r, q\delta^2 > 0$. Given the operator $\boldsymbol{\mathcal{S}} : H^1 \longrightarrow (H^1)'$ defined as $(\xi, \boldsymbol{\mathcal{S}}x) := a(\xi, x)$, the equation (173) is equivalent to $(\xi, \boldsymbol{\mathcal{S}}x) = q\delta\langle \xi, z\rangle + r\langle \dot{\xi}, u\rangle - r\alpha\langle \xi, u\rangle$. For any function $g \in (H^1)'$ we set $u = 0$ and $z = 1/(q\delta)g$. Then $x = \boldsymbol{\mathcal{S}}^{-1}g$ is given by (175) and (176). Furthermore, we obtain the following estimates for the norm of $\boldsymbol{\mathcal{S}}$ and its inverse. Here we employ the H^1–norm $\|x\|_{H^1}^2 = \|x\|_{L_2}^2 + \|\dot{x}\|_{L_2}^2$.

Theorem 4.2 *The operator $\boldsymbol{\mathcal{S}}$ is a linear isomorphism and we have:*

$$\begin{aligned} \min(4r, q\delta^2) \leq \quad & \|\boldsymbol{\mathcal{S}}\|_{H^1 \to (H^1)'} && \leq 2r\max(1, \alpha^2) + q\delta^2, && (179)\\ \frac{c(\alpha)}{q\delta^2} \leq \quad & \|\boldsymbol{\mathcal{S}}^{-1}\|_{(H^1)' \to H^1} && \leq \max\left\{\frac{2}{r}, \frac{2\alpha^2+1}{q\delta^2}\right\}, && (180)\end{aligned}$$

with

$$c(\alpha) \approx |\alpha|^{3/2}\sqrt{\frac{e^2-1}{2(e^2+1)}} \quad \textit{for large } |\alpha|. \tag{181}$$

Hence $cond(\boldsymbol{\mathcal{S}}) = \|\boldsymbol{\mathcal{S}}\|_{H^1 \to (H^1)'}\|\boldsymbol{\mathcal{S}}^{-1}\|_{(H^1)' \to H^1}$ is bounded but tends to infinity with $|\alpha| \to \infty$ for fixed regularization parameters r and q.

Proof: The bilinear form a is continuous and coercive since

$$a(\xi, x) \leq r\|\dot{\xi} - \alpha\xi\|_{L_2}\|\dot{x} - \alpha x\|_{L_2} + q\delta^2\|\xi\|_{L_2}\|x\|_{L_2} \leq C_0\|\xi\|_{H^1}\|x\|_{H^1}$$

with $C_0 = 2r\max(1, \alpha^2) + q\delta^2$ and the parallelogram identity yields

$$\begin{aligned} c_0\|x\|_{H^1}^2 &= c_0\|\dot{x} - \alpha x + \alpha x\|_{L_2}^2 + c_0\|x\|_{L_2}^2 \\ &\leq 2c_0\|\dot{x} - \alpha x\|_{L_2}^2 + c_0(2\alpha^2+1)\|x\|_{L_2}^2 \leq a(x, x) \end{aligned}$$

with $c_0 = \min\left\{\frac{r}{2}, \frac{q\delta^2}{2\alpha^2+1}\right\}$. Hence, due to the theorem of Lax Milgram the operator $\boldsymbol{\mathcal{S}} : H^1 \longrightarrow (H^1)'$ is a linear isomorphism and we have the upper bounds of (179) and (180).

For the lower bound in (179) we set $\bar{x} = \exp(-\alpha t)$. Then

$$\begin{aligned} \|\boldsymbol{\mathcal{S}}\|_{H^1 \to (H^1)'} &\geq \|\boldsymbol{\mathcal{S}}\bar{x}\|_{(H^1)'}/\|\bar{x}\|_{H^1} \geq a(\bar{x}, \bar{x})/\|\bar{x}\|_{H^1}^2 = \\ &= (4r\|\dot{\bar{x}}\|_{L_2}^2 + q\delta^2\|\bar{x}\|_{L_2}^2)/\|\bar{x}\|_{H^1}^2 \geq \min(4r, q\delta^2). \end{aligned}$$

For the lower bound in (180) we consider the function $\bar{x} = \exp(\alpha t)$. Obviously it holds $\boldsymbol{\mathcal{S}}\bar{x} = q\delta^2\bar{x}$ and hence $\|\boldsymbol{\mathcal{S}}^{-1}\|_{(H^1)' \to H^1} \geq \frac{c(\alpha)}{q\delta^2}$ with $c(\alpha) := \|\bar{x}\|_{H^1}/\|\bar{x}\|_{(H^1)'}$.

The involved norms of $\bar{x}$ are:

$$\begin{aligned} \|\bar{x}\|^2_{H^1} &= (1+\alpha^2)\|\bar{x}\|^2_{L_2} = \frac{(1+\alpha^2)}{(2\alpha)}(e^{2\alpha}-1), \\ \|\bar{x}\|^2_{(H^1)'} &= \frac{(2\alpha+1)(\alpha-1)^2(e^{2\alpha}-1)(e^1-e^{-1})+4\alpha^3e^{-1}(e^\alpha-e^1)^2}{2\alpha(\alpha+1)^2(\alpha-1)^2(e^1-e^{-1})}. \end{aligned}$$

Here $\|\bar{x}\|_{(H^1)'} = \sup_{\|v\|^2_{H^1}=1}\int_0^1 v(t)e^{\alpha t}dt$ is determined by the Langrange technique for optimization problems.
The function v has to fulfill $\ddot{v}-v = -\frac{1}{2k}e^{\alpha t}$ with $\dot{v}(0)=\dot{v}(1)=0$, which is true for $v(t) = \frac{1}{2k(\alpha^2-1)}\frac{1}{e^1-e^{-1}}\{\alpha e^t(e^\alpha-e^{-1})+\alpha e^{-t}(e^\alpha-e^1)-e^{t\alpha}(e^1-e^1)\}$. Furthermore, k is given by $\|v\|^2_{H^1}=1$. Then $\|\bar{x}\|_{(H^1)'} = 2k\langle v, v-\ddot{v}\rangle = 2k\|v\|^2_{H^1} = 2k$. For large $|\alpha|$ we conclude $c(\alpha) \approx |\alpha|^{3/2}\sqrt{\frac{e^2-1}{2(e^2+1)}}$. q.e.d.

This indicates clearly that regularization with r is sufficient to obtain a bounded inverse. However, for large $|\alpha|$ the norm is still very large independently of r.

Obviously $\exp(\alpha t)$ is an eigenfunction for $\boldsymbol{\mathcal{S}}$ with the corresponding eigenvalue $q\delta^2$. The next Lemma gives more information about the eigenvalues and eigenfunctions of $\boldsymbol{\mathcal{S}}$.

Lemma 4.1 *The operator $\boldsymbol{\mathcal{S}}$ has infinite, countable many eigenvalues $\hat{\lambda}_i$ and $\hat{\lambda}_i$ behaves like i^2 for large i. The corresponding eigenfunctions $\hat{x}_i$ have i zeros in $[0,1]$. Considering the operator $\boldsymbol{\mathcal{S}}$ for $r=1$ and $q\delta^2=0$ with its eigenvalues $\tilde{\lambda}_i(\alpha)$, we have $\hat{\lambda}_i = r\tilde{\lambda}_i(\alpha)+q\delta^2$ and the eigenfunctions $\hat{x}_i$ depend only on α.*

The smallest eigenvalue is $\hat{\lambda}_0 = q\delta^2$ and the eigenfunction is given by $\hat{x}_0 = \bar{x} = \exp(\alpha t)$.

Proof: The operator $\boldsymbol{\mathcal{S}}$ is defined correspondingly to (174). Hence we obtain the Sturm–Liouville's eigenvalue problem (see e.g. [187]) $r\ddot{x}-(r\alpha^2+q\delta^2)\,x+\lambda x=0$ with $\dot{x}(1)-\alpha x(1)=\dot{x}(0)-\alpha x(0)=0$. This yields the first statements.

Since $\bar{x}$ has no zeros, we can conclude that it is the eigenfunction to the smallest eigenvalue $\hat{\lambda}_0$, which therefore equals $q\delta^2$. q.e.d.

Consequently, the parameters q and δ determine the smallest eigenvalue, r controls the gap between the eigenvalues and α determines the behaviour of the eigenvalues for growing i and gives the corresponding eigenfunctions. Moreover, $\exp(\alpha t)$ is the eigenfunction to the smallest eigenvalue. This is one essential ingredient to construct an efficient preconditioner of the corresponding discretized system (see [32]) and leads to a drastically reduced condition number of the discretized system.

4.2 Linear model with several state functions

In case of a linear model with constant coefficients with several state functions we define correspondingly to (173) and (178) the symmetric bilinear form

$$a(\boldsymbol{\zeta},\boldsymbol{x}) \;=\; \langle\dot{\boldsymbol{\zeta}}-\boldsymbol{A}\boldsymbol{\zeta},\boldsymbol{R}^{-1}(\dot{\boldsymbol{x}}-\boldsymbol{A}\boldsymbol{x})\rangle+\langle\boldsymbol{C}\boldsymbol{\zeta},\boldsymbol{Q}\boldsymbol{C}\boldsymbol{x}\rangle \tag{182}$$

and the operator $\boldsymbol{\mathcal{S}}$ given by $(\boldsymbol{\zeta},\boldsymbol{\mathcal{S}}\boldsymbol{x}) := a(\boldsymbol{\zeta},\boldsymbol{x})$. We cannot extend in a straight forward way the results concerning the eigenvalues. However, we can extend some ideas of Theorem 4.2 taking into account the observability of the system.

Proposition 4.1 *The symmetric bilinear form $a(\boldsymbol{\zeta},\boldsymbol{x})$ is continuous and coercive.*

Proof: We obtain $a(\boldsymbol{\zeta},\boldsymbol{x}) \le \left(2\|\boldsymbol{R}\|^{-1}\max(1,\|\boldsymbol{A}\|^2)+\|\boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}\|\right)\|\zeta\|_{H^1}\|\boldsymbol{x}\|_{H^1}$ as in Theorem 4.2 which yields continuity. To show that a is coercive we first proof positive definiteness. Given $a(\boldsymbol{x},\boldsymbol{x})=0$ then $\boldsymbol{x}$ is the solution of the system $\dot{\boldsymbol{x}}-\boldsymbol{A}\boldsymbol{x}\equiv\boldsymbol{0}$, $\boldsymbol{x}(0)=\boldsymbol{x}_0$ and $(\boldsymbol{y}\equiv)\boldsymbol{C}\boldsymbol{x}\equiv\boldsymbol{0}$. Since the system is observable $\boldsymbol{y}\equiv\boldsymbol{0}$ yields $\boldsymbol{x}_0=\boldsymbol{0}$ and consequently $\boldsymbol{x}\equiv\boldsymbol{0}$. Hence, $a(\boldsymbol{x},\boldsymbol{x})>0$ for $\boldsymbol{x}\not\equiv 0$. Assuming that a is not coercive, there exists a sequence $\{\boldsymbol{x}_n\}\in H^1$ with $\|\boldsymbol{x}_n\|_{H^1}=1$ such that $\lim_{n\to\infty}a(\boldsymbol{x}_n,\boldsymbol{x}_n)=0$. Since $\{\boldsymbol{x}_n\}$ is bounded it has a subsequence which we call for simplicity again $\boldsymbol{x}_n$ and which is weakly convergent in H^1, strongly convergent in L_2 to $\boldsymbol{0}\not\equiv\boldsymbol{x}\in H^1$ and $\|\dot{\boldsymbol{x}}\|_{L_2}\le\lim_{n\to\infty}\|\dot{\boldsymbol{x}}_n\|_{L_2}$. Hence, we obtain the contradiction $0=\lim_{n\to\infty}a(\boldsymbol{x}_n,\boldsymbol{x}_n)\ge a(\boldsymbol{x},\boldsymbol{x})>0$ and coercivity holds. q.e.d.

Corollary 4.1 *For any $\boldsymbol{z},\boldsymbol{u}\in L_2$ the solution $\boldsymbol{x}$ of (173) determines the unique solution of the minimization problem (171).*

For the corresponding operator $\boldsymbol{\mathcal{S}}$ and its inverse we obtain the following specific estimates depending on the model matrices and the regularization matrices.

Theorem 4.3 *The operator $\boldsymbol{\mathcal{S}}: H^1\to(H^1)'$ is a linear isomorphism and has a bounded inverse. Furthermore, we have*

$$\|\boldsymbol{\mathcal{S}}\|_{H^1\to(H^1)'}\le 2\|R\|^{-1}\max(1,\|\boldsymbol{A}\|^2)+\|\boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}\|, \tag{183}$$

and

$$\max\{k_S(\boldsymbol{v})|\boldsymbol{v}\text{ eigenvector of }\boldsymbol{A}\text{ with }\|v\|_{l_2}=1\} \;\le\; \|\boldsymbol{\mathcal{S}}\|_{H^1\to(H^1)'}, \tag{184}$$

$$\max\{c(\alpha)k_{S^{-1}}(\boldsymbol{v})|\alpha,\boldsymbol{v}\text{ with }\boldsymbol{A}\boldsymbol{v}=\alpha\boldsymbol{v}\text{ with }\|v\|_{l_2}=1\} \;\le\; \|\boldsymbol{\mathcal{S}}^{-1}\|_{(H^1)'\to H^1} \tag{185}$$

with

$$k_S(\boldsymbol{v}) \;=\; \min\{4(\boldsymbol{v}^T\boldsymbol{R}^{-1}\boldsymbol{v}),(\boldsymbol{v}^t\boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}\boldsymbol{v})\}>0, \tag{186}$$

$$\infty>k_{S^{-1}}(\boldsymbol{v}) \;=\; 1/\|\boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}\boldsymbol{v}\|_{l_2}>1/\|\boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}\|>0 \tag{187}$$

and $c(\alpha)$ given as in Theorem 4.2.

If the spectral radius $\rho(\boldsymbol{A})$ of $\boldsymbol{A}$ is larger than 1, we can choose

$$k_S(\boldsymbol{v}) = 2\boldsymbol{v}^T\boldsymbol{R}^{-1}\boldsymbol{v} \geq 2/\|\boldsymbol{R}\|. \tag{188}$$

For any fixed regularization parameters $\boldsymbol{R}$ and $\boldsymbol{Q}$, $cond(\boldsymbol{\mathcal{S}}) \geq c(\alpha)k_S(\boldsymbol{v})k_{S^{-1}}(\boldsymbol{v})$ can be arbitrarily large, if the spectral radius $\rho(\boldsymbol{A})$ of $\boldsymbol{A}$ is large or if there is an eigenvector $\boldsymbol{v}$ of $\boldsymbol{A}$ which is close to the null space of $\boldsymbol{C}$.

Proof: Due to the Riesz representation theorem and Proposition 4.1 $\boldsymbol{\mathcal{S}}$ is an linear isomorphism with an bounded inverse and the estimation (183) holds. For the other estimates let α be an eigenvalue of $\boldsymbol{A}$ and $\boldsymbol{v}$ a normalized corresponding eigenvector. Then, defining $\bar{\boldsymbol{x}} = exp(-\alpha t)\boldsymbol{v}$ we obtain

$$a(\bar{\boldsymbol{x}}, \bar{\boldsymbol{x}}) = 4\|\dot{\bar{\boldsymbol{x}}}\|_{L_2}^2(\boldsymbol{v}^T\boldsymbol{R}^{-1}\boldsymbol{v}) + \|\bar{\boldsymbol{x}}\|_{L_2}^2(\boldsymbol{v}^t\boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}\boldsymbol{v}).$$

Following the steps in Theorem 4.2 we have $k_S(\boldsymbol{v}) < \|\boldsymbol{\mathcal{S}}\|_{H^1\to(H^1)'}$ for any eigenvector $\boldsymbol{v}$. To show $k_S(\boldsymbol{v}) > 0$ we use the observability condition: We consider the model equations $\dot{\boldsymbol{x}} - \boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}$, $\boldsymbol{x}(0) = \boldsymbol{x}_0$ and $\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x}$, which is required to be observable. Hence $\boldsymbol{x}_0 \mapsto \boldsymbol{y}$ is injective. Therefore defining $\boldsymbol{x}_0 := \boldsymbol{v} \neq \boldsymbol{0}$ we follow $exp(\alpha t)\boldsymbol{C}\boldsymbol{v} = \boldsymbol{y} \not\equiv \boldsymbol{0}$ and thus $\boldsymbol{C}\boldsymbol{v} \neq 0$. Then $\boldsymbol{v}^t\boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}\boldsymbol{v} = \|\boldsymbol{Q}^{1/2}\boldsymbol{C}\boldsymbol{v}\|^2 \geq \|\boldsymbol{C}\boldsymbol{v}\|^2/\|\boldsymbol{Q}^{-1}\| > 0$ and $k_S(\boldsymbol{v}) > 0$. Since $\boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}\boldsymbol{v} \neq 0$ we also have $\infty > k_{S^{-1}}(\boldsymbol{v})$.

If the spectral radius of $\boldsymbol{A}$ is larger or equal than 1, there exists an eigenvalue $\alpha > 1$ and we can estimate

$$\|\boldsymbol{\mathcal{S}}\|_{H^1\to(H^1)'} \geq \frac{a(\bar{\boldsymbol{x}}, \bar{\boldsymbol{x}})}{\|\bar{\boldsymbol{x}}\|_{H^1}^2} = \frac{4\alpha^2(\boldsymbol{v}^T\boldsymbol{R}^{-1}\boldsymbol{v}) + (\boldsymbol{v}^t\boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}\boldsymbol{v})}{(1+\alpha^2)} \geq 2\boldsymbol{v}^T\boldsymbol{R}^{-1}\boldsymbol{v}$$

which gives (188).

For estimation (185) we define $\bar{\boldsymbol{x}} = \exp(\alpha t)\boldsymbol{v}$, which yields $(\boldsymbol{\xi}, \boldsymbol{\mathcal{S}}\bar{\boldsymbol{x}}) = \langle\boldsymbol{\xi}, \boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}\bar{\boldsymbol{x}}\rangle$. Thus we can follow for any $\alpha, \boldsymbol{v}$

$$\frac{1}{\|\boldsymbol{C}^T\boldsymbol{Q}\boldsymbol{C}\boldsymbol{v}\|_{l^2}}\, c(\alpha) \leq \|\boldsymbol{\mathcal{S}}^{-1}\|_{(H^1)'\to H^1}.$$

q.e.d.

While the boundedness of the inverse $\boldsymbol{\mathcal{S}}^{-1}$ is known, a specified upper bound is missing. However, in an analogous manner to the case of one state function we obtain:

Proposition 4.2 *For an invertible matrix $\boldsymbol{C}$ the inverse operator $\boldsymbol{\mathcal{S}}^{-1}$ is bounded by*

$$\|\boldsymbol{\mathcal{S}}^{-1}\|_{(H^1)'\to H^1} \leq \max\{2\|\boldsymbol{R}\|, (2\|\boldsymbol{A}\|^2 + 1)\|\boldsymbol{Q}^{-1}\|\|\boldsymbol{C}^{-1}\|^2\}.$$

Due to the boundedness of the inverse and the compact inbedding $H^1 \hookrightarrow C^0$ we have:

Corollary 4.2 *The regularized problem formulation (150) with time independent, linear ODE's as constraints and no inequality constraints is well posed, i.e.* $\max\{|x_0|, \|\boldsymbol{w}\|_{L_2}\} \leq c\|\boldsymbol{z}\|_{L_2}$. *Furthermore we have* $\|\boldsymbol{x}\|_{C^0} \leq c\|\boldsymbol{z}\|_{L_2} \leq c\|\boldsymbol{z}\|_{L_\infty}$ *with a generic constant c.*

As for one state only, Theorem 4.3 shows that independent of the regularization of $\boldsymbol{w}$ we face for a large spectral radius $\rho(\boldsymbol{A})$ a high condition number. This influences, for example, strongly the condition number relevant for the efficiency of the iterative solver for the discretized system corresponding to (173), see for example [26, 32], and the accuracy of the solution with respect to perturbations.

To be more precise concerning the constants involved in the estimates in [26], there we considered the not completely reduced system for $\boldsymbol{x}$, $\boldsymbol{\lambda}$ and $\hat{\boldsymbol{\lambda}} = -\boldsymbol{\lambda}_2$ (see (151)) which yields the saddle point problem

$$\left(\boldsymbol{\nu}, \tilde{\mathcal{L}}\begin{pmatrix}\boldsymbol{\lambda}\\ \hat{\boldsymbol{\lambda}}\\ \boldsymbol{x}\end{pmatrix}\right) := \begin{pmatrix}\langle \boldsymbol{Q}^{-1}\hat{\boldsymbol{\lambda}} + \boldsymbol{C}\boldsymbol{x}, \boldsymbol{\nu}_1\rangle\\ \langle \boldsymbol{R}\boldsymbol{\lambda} - \dot{\boldsymbol{x}} + \boldsymbol{A}\boldsymbol{x}, \boldsymbol{\nu}_2\rangle\\ \langle \boldsymbol{\lambda}, -\dot{\boldsymbol{\nu}}_3 + \boldsymbol{A}\boldsymbol{\nu}_3\rangle + \langle \boldsymbol{C}^T\hat{\boldsymbol{\lambda}}, \boldsymbol{\nu}_3\rangle\end{pmatrix} = \langle\begin{pmatrix}\boldsymbol{z}\\ -\boldsymbol{u}\\ 0\end{pmatrix}, \boldsymbol{\nu}\rangle. \quad (189)$$

The first equation reflects (151) together with (154). The relevant constants are c_7 and c_8 with

$$c_7(\|\boldsymbol{\lambda}\|_{L_2} + \|\hat{\boldsymbol{\lambda}}\|_{L_2} + \|\boldsymbol{x}\|_{H^1}) \leq \|\tilde{\mathcal{L}}(\boldsymbol{\lambda}, \hat{\boldsymbol{\lambda}}, \boldsymbol{x})\|_{L_2\times L_2\times(H^1)'} \leq c_8(\|\boldsymbol{\lambda}\|_{L_2} + \|\hat{\boldsymbol{\lambda}}\|_{L_2} + \|\boldsymbol{x}\|_{H^1})$$

(see [26] equation (119)).

For $\bar{\boldsymbol{x}} = \exp(\alpha t)\boldsymbol{v}$ we have $\|\tilde{\mathcal{L}}(\boldsymbol{0}, \boldsymbol{0}, \bar{\boldsymbol{x}})\|_{L_2\times L_2\times(H^1)'} = \|\boldsymbol{C}\boldsymbol{v}\|\|\exp(\alpha t)\|_{L_2}$. Hence we can follow

$$c_7 \leq \|\boldsymbol{C}\boldsymbol{v}\|\|\exp(\alpha t)\|_{L_2}/\|\exp(\alpha t)\|_{H^1} = \|\boldsymbol{C}\boldsymbol{v}\|/\sqrt{1+\alpha^2}.$$

For $\bar{\boldsymbol{x}} = \exp(-\alpha t)\boldsymbol{v}$ we have $\|\tilde{\mathcal{L}}(\boldsymbol{0}, \boldsymbol{0}, \bar{\boldsymbol{x}})\|^2_{L_2\times L_2\times(H^1)'} = (\|\boldsymbol{C}\boldsymbol{v}\|^2 + 4\alpha^2)\|\exp(-\alpha t)\|_{L_2}$ and

$$c_8 \geq \sqrt{\|\boldsymbol{C}\boldsymbol{v}\|^2 + 4\alpha^2}/\sqrt{1+\alpha^2}.$$

Consequently

$$cond(\tilde{\mathcal{L}}) = c_8/c_7 \geq \sqrt{1 + 4\frac{\alpha^2}{\|\boldsymbol{C}\boldsymbol{v}\|^2}}$$

for all eigenvalues α and corresponding eigenvectors $\boldsymbol{v}$. Hence independent of the regularization parameters $cond(\tilde{\mathcal{L}})$ tends to infinity if the spectral radius of $\boldsymbol{A}$ tends to infinity or if there is an eigenvector which tends to the null space of $\boldsymbol{C}$.

Finally, we would like to bring to the readers attention the influence of a long time horizon. As mentioned in the beginning state estimation uses measurements of a past time horizon $[0, T]$, which is then projected onto $[0, 1]$. Given a fixed model equation $\dot{\boldsymbol{x}}(t) - \boldsymbol{A}\boldsymbol{x}(t) - \boldsymbol{u}(t) = \boldsymbol{0}$ for $t \in [0, T]$ the projection onto $[0, 1]$ leads to $\dot{\tilde{\boldsymbol{x}}}(s) - T\boldsymbol{A}\tilde{\boldsymbol{x}}(s) - \tilde{\boldsymbol{u}}(s) = \boldsymbol{0}$ for $s \in [0, 1]$. Hence in the estimation for $cond(\boldsymbol{\mathcal{S}})$ the spectral radius $T\rho(\boldsymbol{A})$ is relevant.

Remark 4.4 *While long horizons may be favourable for stability reasons concerning the confidence into the obtained measurements, from the point of view of the resulting system long horizons should be avoided, since they increase the condition number.*

Some first approaches for the choice of an appropriate length of the time horizon are known under stochastical assumptions on the underlying measurement errors and for linear systems [138, 144, 156]. However this is still a fairly open research problem, in particular if stochastical assumptions cannot be assured in the applications or nonlinearity occurs. The above remark ensures the importance of closing this gap and illuminates an additional issue which should to be considered for the choice of the horizon length.

5 Conclusion

The formulation of state estimation as an optimization problem including possible model error functions but without regularization of the initial data offers solutions without bias. It turned out that the inclusion of model error functions into the output equations is not necessary for the computation process. In this way the otherwise difficult task of choosing an appropriate regularization parameter for these model error function can be avoided. Moreover, one can reduce the resulting optimization problem without disadvantage to the unknown states and the model error functions in the state equations.

For linear ODE's with time independent coefficients and without inequality constraints it is shown that the condition numbers of the evolving operators can be arbitrarily large independently of the chosen regularization parameters, if the spectral radius of the system matrix is large or an eigenvector of the system matrix is close to the null space of the matrix, which depicts some states as measurable. These condition numbers influence for example essentially the numerical solution process. Furthermore, in the one dimensional case the eigenfunction to the smallest eigenvalue of the operator is derived, which is one important ingredient to obtain an efficient preconditioner for the numerical approach in [32].

Preconditioning via a Schur Complement Method: an Application in State Estimation[6]

Luise Blank

to be published in: SIAM Journal on Scientific Computing

Abstract

In this work the Schur complement is analyzed and applied for preconditioning symmetric matrices. In particular we eliminate the in modulo extreme eigenvalues by computing the Schur complement with respect to a coarse subspace, which captures the corresponding eigenvectors. The potential of this method is presented as an application in state estimation, where we employ a multiscale approach based on a wavelet-Galerkin discretization. The hierarchy of arising optimization problems is solved in the framework of nested iterations. Due to the wavelet approach, for all scales we can use one coarse adapted subspace in the Schur complement method. Typically this provides not only a reduced condition number but also large gaps in the arising eigenvalue distribution, which is favorable for several iterative methods. Numerical investigations show the enormous reduction in the condition number, the necessary cg-iteration steps and, consequently, in the computation time.

Key Words: Schur complement, preconditioning, deflation, state estimation, multiscale approach, wavelets.

1 Introduction

The most common Schur complement technique applies the Schur complement reduction repeatedly until the dimension of the Schur complement is small enough for fast inversion. The reduction in dimension is a key feature. However, it requires that the inversions in each reduction step to be performed efficiently. Typically one searches for a subspace in which the corresponding matrix is block diagonal, well conditioned and not too small.

In contrast, in this paper we would like to present an application of the Schur complement method in which we apply the reduction step only once. We look for a

[6]This work has been supported by the "Deutsche Forschungsgemeinschaft" under grant MA1188/6.

subspace in which the corresponding matrix is probably dense, as small as possible, but captures the ill-conditioning of the matrix. Consequently, the dimension of the Schur complement remains large, but the goal is to reduce the condition number drastically in comparison to the original matrix. Hence the Schur complement is used for eigenvalue deflation.

The potential of the method is presented as an application in state estimation. For better motivation let us briefly describe the general framework in which we apply the method. In chemical engineering complex dynamic optimization problems have to be solved online. In [20, 28] a multiscale approach based on wavelets is presented in which a hierarchy of successively, adaptively refined problems is constructed. These problems are solved in the framework of nested iteration as long as the real-time restrictions are fulfilled. Each discrete problem has to be solved only with an accuracy comparable to the current approximation error. Hence the use of an iterative solver also suggests itself for the arising systems of linear equations. Obviously the eigenvalue distribution plays an important role in the efficiency. In [26] it is shown that by using properly scaled wavelets the condition numbers can be bounded independently of the refinement level. However, the in modulo smallest eigenvalues are still close to zero due to inherent features of the estimation problem, i.e., the original ill-posedness, the low measure of observability, and the stiff involved differential equations.

A number of articles have established the benefits of eigenvalue deflation when using Krylov subspace methods. Most common is the deflation of nearly invariant subspaces. Some authors augment the Krylov subspace with approximate eigenvectors that either are obtained during the iteration [140, 159] or are known a priori [183]. Others use subdomain deflation, which is related to domain decomposition [80, 130, 147]. The Schur complement fits into the framework of the projection used in this technique of eigenvalue deflation. However, in general it does not employ invariant subspaces. To distinguish mesh dependency of eigenvalues, Schur complement techniques are used, for example, as saddle point preconditioners for discrete Navier–Stokes equations [75] or elliptic problems with Neumann boundaries (see [15] and references therein). Using the Schur complement to eliminate negative eigenvalues of indefinite but symmetric systems arising in nonlinear elasticity is considered in [164] and is based on the Sylvester law of inertia.

We apply the above Schur complement preconditioning in order to eliminate the smallest eigenvalues. The idea is based on the following: roughly speaking, if we compute the Schur complement with respect to a subspace containing the eigenvectors corresponding to the smallest eigenvalues, then these eigenvalues are eliminated in the Schur complement. To approximate such a subspace we consider the underlying infinite dimensional, but wavelet discretized, system. We try to find a very coarse, but adapted, set of wavelet basis functions that captures the eigenfunctions (and with them the corresponding eigenvectors) to the smallest eigenvalues. Due to the multiscale representation, one expects that this coarse wavelet space will also capture good approximations of the eigenvectors to the smallest eigenvalues of the truncated systems. Then, for the whole hierarchy of systems, the Schur complements with respect to this coarse, fixed subspace will give rise to well-conditioned systems.

The outline of the paper is as follows. In section 2 the formulation of the dynamic state estimation is briefly introduced. Furthermore wavelets and their basic properties are described in order to motivate their application in this context. Then we present the wavelet-Galerkin discretization for the multiscale approach, which leads to the sequence of linear equation systems that we will study. Wavelet-based preconditioning eliminates the scale-to-scale increase of the large eigenvalues. We conclude this section by commenting on the (in our application) typically arising eigenvalue distributions and their origins. The analytical results presented in section 3 provide the foundation of the application of the Schur complement method as a preconditioner. As a first step, in section 3.1 we deduce estimations for eigenvalues of symmetric preconditioned matrices with respect to the eigenvalues of the original matrix. Using the fact that the Schur complement arises by a particular symmetric preconditioning, we prove under certain assumptions the elimination of the extreme eigenvalues. The choice of an appropriate subspace is then motivated in section 3.2. In section 4 the nested iteration algorithm based on the Schur complement technique is sketched and its application is numerically investigated. We illustrate the drastic effect on the eigenvalue distribution and, hence, on the condition number and iteration number. In section 4.4 we analyze the influence of the choice of the subspace in the numerical results. Finally, in section 5 we summarize the conclusions.

2 Problem Formulation and Discretization

2.1 Problem Formulation

Dynamic state estimation aims to determine, from a few given, noisy measurements, all states involved in the process. The dynamic behavior of the plant is often modeled by a system of differential-algebraic equations for all states $\boldsymbol{x}$ depending on given controls $\boldsymbol{u}$. Moreover, unknown model error functions $\boldsymbol{w}$ are included in the model equations to represent unstructured model uncertainty. Typically only very few states can be measured, and these states are denoted by the output functions $\boldsymbol{y}$. Then state estimation can be formulated as a constrained optimization problem, where the difference between measured data $\boldsymbol{z}$ and corresponding modeled outputs $\boldsymbol{y}$ has to be minimized [6, 20]. Together with bounds on selected variables, the model equations form the constraints of the optimization. We stress that the problem in its original form is an ill-posed inverse problem. Therefore some regularization is necessary [28, 29].

Since we concentrate on the solution of the final linear systems, we consider here only linear model problems without inequality constraints. The estimation horizon is scaled to $[0,1]$. Hence, we focus here on the following particular setting, where we suppress for convenience explicit references to the domain $[0,1]$. For given $\boldsymbol{z} \in (L_2)^{n_y}$ and $\boldsymbol{u} \in (L_2)^{n_u}$ and unknown functions $\boldsymbol{x} \in (H^1)^{n_x}$, $\boldsymbol{w} \in (L_2)^{n_w}$, and $\boldsymbol{y} \in (L_2)^{n_y}$, the resulting optimization problem has the form

$$1/2 \min_{\boldsymbol{x}(0),\boldsymbol{w}} \left\{ ||\boldsymbol{y} - \boldsymbol{z}||^2_{L_2^{n_y}} + \varepsilon \, ||\boldsymbol{w}||^2_{L_2^{n_w}} \right\} \tag{190}$$

subject to the constraints

$$\begin{aligned} \dot{\boldsymbol{x}} - \boldsymbol{A}\boldsymbol{x} - \boldsymbol{W}\boldsymbol{w} &= \boldsymbol{u}, & (191)\\ \boldsymbol{y} - \boldsymbol{C}\boldsymbol{x} &= \boldsymbol{0}. & (192) \end{aligned}$$

Even for nonlinear model equations, (192) remains typically linear, since $\boldsymbol{C}$ selects some states as outputs. Consequently, $\boldsymbol{y}$ can be eliminated in the optimization problem. We would like to stress that in the above problem the initial data $\boldsymbol{x}(0)$ are not given but have to be estimated. They are also not included in the cost functional as is often seen in other applications. This has a major impact on the condition number. In addition, the observability measure of the model [110, 185] represented by the constraints, the stiffness of the differential equations, the incorporated model error functions, and the choice of the regularization parameter ε influences the condition of the formulated optimization problem, and hence the numerical treatment of the resulting linear systems of equations.

If we denote the Lagrange multipliers by $\boldsymbol{\lambda}$, the resulting first order necessary conditions are the following [98]:

$$\begin{aligned} \varepsilon\boldsymbol{w} - \boldsymbol{W}^T\boldsymbol{\lambda} &= \boldsymbol{0}, & (193)\\ \dot{\boldsymbol{x}} - \boldsymbol{A}\boldsymbol{x} - \boldsymbol{W}\boldsymbol{w} &= \boldsymbol{u}, & (194)\\ -\dot{\boldsymbol{\lambda}} - \boldsymbol{A}^T\boldsymbol{\lambda} + \boldsymbol{C}^T\boldsymbol{C}\boldsymbol{x} &= \boldsymbol{C}^T\boldsymbol{z}, & (195)\\ \boldsymbol{\lambda}(0) = \boldsymbol{0} \qquad \text{and} \qquad \boldsymbol{\lambda}(1) &= \boldsymbol{0}. & (196) \end{aligned}$$

Obviously $\boldsymbol{w}$ can be eliminated in (194) by the Lagrange multipliers $\boldsymbol{\lambda}$ using (193). Defining $\boldsymbol{R} := 1/\varepsilon \boldsymbol{W}\boldsymbol{W}^T$, we see that the weak formulation of the remaining equations is given by

$$\begin{aligned} \langle \dot{\boldsymbol{x}} - \boldsymbol{A}\boldsymbol{x} - \boldsymbol{R}\boldsymbol{\lambda}, \boldsymbol{\zeta}_1 \rangle &= \langle \boldsymbol{u}, \boldsymbol{\zeta}_1 \rangle \text{ for all } \boldsymbol{\zeta}_1 \in L_2^{n_x}, & (197)\\ \langle \boldsymbol{\lambda}, \dot{\boldsymbol{\zeta}}_2 \rangle + \langle -\boldsymbol{A}^T\boldsymbol{\lambda} + \boldsymbol{C}^T\boldsymbol{C}\boldsymbol{x}, \boldsymbol{\zeta}_2 \rangle &= \langle \boldsymbol{C}^T\boldsymbol{z}, \boldsymbol{\zeta}_2 \rangle \text{ for all } \boldsymbol{\zeta}_2 \in H_1^{n_x}. & (198) \end{aligned}$$

Since our discretization is based on a Galerkin discretization of (197)–(198) using wavelets, we briefly introduce their basic properties and indicate their significance in the present context. For a more detailed treatment of wavelets we refer the reader to [43, 56, 64].

2.2 Wavelets

Apart from boundary adaptation wavelet functions $\psi_{j,k}$ are obtained by dilation and translation of a suitable *mother wavelet* ψ, i.e., $\psi_{j,k} = 2^{j/2}\psi(2^j \cdot -k)$, where the normalizing factor $2^{j/2}$ keeps the L_2-norm of the $\psi_{j,k}$ independent of j,k. Thus when ψ is chosen to have bounded support, as it will always be the case in our applications, one has *diam supp* $\psi_{j,k} \sim 2^{-j}$, $j \to \infty$. One of the key features of wavelets is that for suitable ψ the collection $\mathbf{\Psi} = \{\psi_{j,k}\}$ forms a *Riesz basis* of L_2; i.e., any $f \in L_2$ has a unique expansion

$$f = \mathbf{\Psi}^T \boldsymbol{d}_f = \sum_{j,k} d_{j,k}\psi_{j,k},$$

and the norm equivalence

$$c_1 \parallel \boldsymbol{d}_f \parallel_{\ell_2} \leq \parallel f \parallel_{L_2} \leq c_2 \parallel \boldsymbol{d}_f \parallel_{\ell_2} \tag{199}$$

holds for some positive constants c_1, c_2 independent of f. Clearly (199) means that small perturbations of the expansion coefficients $\boldsymbol{d}_f$ cause only small changes of f in the L_2-norm and vice versa. Depending on the smoothness of the employed wavelets, norm equivalences with respect to other function spaces such as Sobolev and Besov spaces also hold with appropriate scale-dependent weights on the wavelet coefficients. They are, for example, essential for signal analysis and for the preconditioning of discretized differential (see section 2.4) and integral operators.

In general, (199) implies the existence of a *dual basis* $\tilde{\mathbf{\Psi}}$ in L_2 that is *biorthogonal* to $\mathbf{\Psi}$, i.e., $\langle\langle \mathbf{\Psi}, \tilde{\mathbf{\Psi}} \rangle\rangle = \boldsymbol{I}$, where we use the shorthand notation $\langle\langle \mathbf{\Psi}, \tilde{\mathbf{\Psi}} \rangle\rangle = (\langle \psi_{j,k}, \tilde{\psi}_{i,l} \rangle)_{(j,k),(i,l)}$ with $\langle \cdot, \cdot \rangle$ denoting the standard L_2-inner product. Thus the wavelet coefficients $d_{j,k}$ are given by $\langle f, \tilde{\psi}_{j,k} \rangle$.

The starting point for the construction of wavelet bases is usually a so-called *multiresolution sequence* of nested spaces $S_{j_0} \subset \cdots \subset S_j \subset S_{j+1} \subset \cdots$ whose union is dense in L_2. Here j_0 stands for some coarsest discretization level. The spaces S_j are spanned by scaling functions, $S_j = \operatorname{span}\{\varphi_{j,k} : k \in \mathcal{I}_j\}$, which are obtained by a suitable, compactly supported function φ. Due to the compact support of φ we can view 2^{-j} as the uniform mesh size of S_j. Given a suitable dual sequence $\tilde{S}_j$, a successive decomposition of S_j leads to the multiscale splitting $S_J = S_{j_0} \bigoplus_{j=j_0}^{J-1} W_j$, where $W_j = \operatorname{span}\{\psi_{j,k} : k \in \mathcal{I}'_j\} \perp \tilde{S}_j$. Analogous splittings for the spaces $\tilde{S}_j$ yield complement spaces $\tilde{W}_j \perp S_j$. The bases $\mathbf{\Psi}$ and $\tilde{\mathbf{\Psi}}$ for the spaces W_j and $\tilde{W}_j$, respectively, are biorthogonal.

All the above properties can be realized when choosing φ as the cardinal B-spline of order m. Moreover, one can construct for any $\tilde{m} \in I\!N, \tilde{m} \geq m, m+\tilde{m}$ even, a compactly supported dual scaling function $\tilde{\varphi}$ such that the dual multiresolution spaces $\tilde{S}_j$ contain all polynomials up to order $\tilde{m}$. As a consequence of biorthogonality, one has the *moment conditions* $\langle (\cdot)^r, \psi \rangle = 0$ for $r = 0, \ldots, \tilde{m}-1$. Likewise one has $\langle (\cdot)^r, \tilde{\psi} \rangle = 0$ for $r = 0, \ldots, m-1$. Therefore smooth functions have wavelet expansions with rapidly decaying wavelet coefficients so that scalewise truncation of wavelet expansions provides good approximations. However, as soon as sharp transitions or even singularities occur, different nonuniform selections of basis functions are expected to provide more economical approximations. In fact, combining the above statements with the locality of wavelets, large wavelet coefficients reflect a large local change of the function. In particular, retaining only the first N largest coefficients in modulus provides —up to a uniform constant— the best approximation to f that can be composed from any selection of N basis functions (*best N-term approximation*). This is the basis for nonlinear compression techniques. Accordingly, the objective of *adaptive* approximation is to successively track the most significant coefficients of the function that is to be determined.

2.3 Discretization: Wavelets as Trial and Test Functions

The Galerkin discretization of the optimization problem based on wavelets is given by using a representation of each function with respect to an appropriately chosen wavelet basis in the first order necessary conditions (197)–(198). In our case we use piecewise linear, continuous wavelets $\boldsymbol{\Psi}$ as trial functions for $\boldsymbol{x}$ and $\boldsymbol{\zeta}_2$ (i.e. the scaling function is a B-spline of order $m = 2$ fulfilling the moment conditions with $\tilde{m} = 2$), and the functions $\boldsymbol{z}$ are discretized by the dual wavelets $\tilde{\boldsymbol{\Psi}}$, while we employ piecewise constant wavelets, so-called Haar wavelets $\boldsymbol{\Psi}^H$ ($m = \tilde{m} = 1$) for $\boldsymbol{u}$, the Lagrange multipliers $\boldsymbol{\lambda}$ and the test functions $\boldsymbol{\zeta}_1$. This corresponds to the minimal regularity for a conforming discretization. Consequently, we use the representations $\boldsymbol{x} = \boldsymbol{\Psi}^T_{\Lambda_x}\boldsymbol{d}_{\boldsymbol{x}}$ and $\boldsymbol{\lambda} = (\boldsymbol{\Psi}^H_{\Lambda_\lambda})^T\boldsymbol{d}_{\boldsymbol{\lambda}}$, etc. in (197)–(198). We obtain a system of equations for the unknown wavelet coefficients $\boldsymbol{d}_{\boldsymbol{x}}$ and $\boldsymbol{d}_{\boldsymbol{\lambda}}$. The discretization matrix can be described in a rather condensed form with the aid of Kronecker products. The Kronecker product of a matrix $\boldsymbol{A} \in R^{n\times m}$ with a matrix $\boldsymbol{B}$ is given by $\boldsymbol{A}\otimes\boldsymbol{B} := (a_{i,j}\boldsymbol{B})_{\substack{i=1,\ldots,n\\ j=1,\ldots,m}}$. Defining

$$\boldsymbol{R}_\Lambda = \boldsymbol{R}\otimes\boldsymbol{I}_\Lambda, \quad \boldsymbol{Q}_\Lambda = \boldsymbol{C}^T\boldsymbol{C}\otimes\langle\langle\boldsymbol{\Psi}_\Lambda,\boldsymbol{\Psi}_\Lambda\rangle\rangle, \quad \boldsymbol{C}_\Lambda = \boldsymbol{C}\otimes\boldsymbol{I}_\Lambda, \tag{200}$$

$$\boldsymbol{D}_\Lambda = \boldsymbol{I}_{n_x}\otimes\langle\langle\boldsymbol{\Psi}^H_{\Lambda'},\dot{\boldsymbol{\Psi}}_\Lambda\rangle\rangle, \quad \boldsymbol{A}_\Lambda = \boldsymbol{A}\otimes\langle\langle\boldsymbol{\Psi}^H_{\Lambda'},\boldsymbol{\Psi}_\Lambda\rangle\rangle, \tag{201}$$

yields the following system of equations, which is equivalent to (197)–(198) and still *infinite dimensional* but *discretized*:

$$\begin{pmatrix} -\boldsymbol{R}_\Lambda & \boldsymbol{D}_\Lambda - \boldsymbol{A}_\Lambda \\ (\boldsymbol{D}_\Lambda - \boldsymbol{A}_\Lambda)^T & \boldsymbol{Q}_\Lambda \end{pmatrix} \begin{pmatrix} \boldsymbol{d}_{\boldsymbol{\lambda}} \\ \boldsymbol{d}_{\boldsymbol{x}} \end{pmatrix} = \begin{pmatrix} \boldsymbol{d}_{\boldsymbol{u}} \\ \boldsymbol{C}^T_\Lambda\boldsymbol{d}_{\boldsymbol{z}} \end{pmatrix}. \tag{202}$$

While the above formulation still provides the exact solution, for the numerical treatment we have to choose finite dimensional approximations. Hence for each function involved we have to choose a finite set of basis functions, described by their index set $\Lambda_{\boldsymbol{x}}$, etc.. As mentioned before, we will start with some initial, possibly small index sets $\Lambda^0_{\boldsymbol{x}}$, etc. and adaptively enlarge these index sets to $\Lambda^{\ell+1}_{\boldsymbol{x}} = \Lambda^\ell_{\boldsymbol{x}}\cup\Gamma^\ell_{\boldsymbol{x}}$ in each refinement step. Moreover, the Haar basis consists of one less scaling function on the coarsest scale but consists of the same number of wavelets. We have $\Lambda' = \Lambda \setminus \{(1,2)\}$. The involved wavelet mass matrices exhibit so-called finger structure with $O(\#\Lambda log(\#\Lambda))$ entries.

Without loss of generality, we consider below only the case $\boldsymbol{W} = \boldsymbol{I}$, i.e., $\boldsymbol{R} = 1/\varepsilon \boldsymbol{I}$, since the following considerations for the iterative solution are extendible to the symmetric indefinite case. Obviously we can eliminate the Lagrange multipliers $\boldsymbol{d_\lambda}$ in (202) and obtain with

$$\boldsymbol{X}_\Lambda \;:=\; \varepsilon\,(\boldsymbol{D}_\Lambda - \boldsymbol{A}_\Lambda)^T\,(\boldsymbol{D}_\Lambda - \boldsymbol{A}_\Lambda) + \boldsymbol{Q}_\Lambda \tag{203}$$

an equation for the wavelet coefficients $\boldsymbol{d_x}$ of the states as follows:

$$\boldsymbol{X}_\Lambda \boldsymbol{d_x} \;=\; \boldsymbol{C}_\Lambda^T \boldsymbol{d_z} + \varepsilon\,(\boldsymbol{D}_\Lambda - \boldsymbol{A}_\Lambda)^T\,\boldsymbol{d_u}. \tag{204}$$

Since (204) is solved iteratively, the matrix $\boldsymbol{X}_\Lambda$, as well as $\boldsymbol{D}_\Lambda, \boldsymbol{A}_\Lambda$, and $\boldsymbol{Q}_\Lambda$, is not assembled. Instead only matrix-vector multiplications are realized. We recall that (204) is solved in the context of nested iteration, i.e., in a sequence of equations for Λ^ℓ, $\ell = 0, 1, \ldots$ with $\Lambda^\ell \subseteq \Lambda^{\ell+1}$ and consequently with increasing dimension.

2.4 Iterative Solver and Eigenvalue Distribution

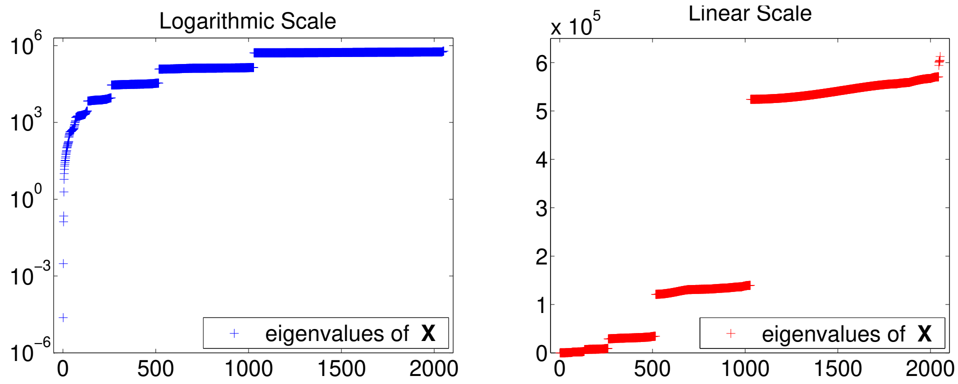
Since $\boldsymbol{X}_\Lambda$ is symmetric and positive definite we apply the pcg-method. The approximate solution $\boldsymbol{d_x^{\ell-1}}$ on $\Lambda^{\ell-1}$ serves as an initial value in the ℓth step. A change of basis is not necessary due to the choice of wavelet basis. The vector $\boldsymbol{d_x^{\ell-1}}$ has only to be extended by zeros for the indices in $\Lambda^\ell/\Lambda^{\ell-1}$. The quality of this initial value depends on the chosen refinement $\Lambda^\ell/\Lambda^{\ell-1}$, which not only determines the change in $\boldsymbol{X}_\Lambda$ but also specifies the additional informations of the signal $\boldsymbol{z}$ and the controls $\boldsymbol{u}$, which have to be processed.

Common knowledge for the pcg-method is an error estimation depending on the condition number $\kappa = \lambda_{max}(\boldsymbol{X})/\lambda_{min}(\boldsymbol{X})$. Results concerning the influence of the distribution of the whole spectrum and, in particular, the influence of large gaps between the largest or smallest eigenvalues, are summarized in [9, 90, 134]. Essentially these results show that after several iterations (depending on the gap) the convergence rate is independent of the extreme eigenvalues. Moreover, it is known that the rate of convergence has the tendency to increase, which may be called superlinear convergence [134, 179]. In [9] the convergence behavior is divided into the following three phases: in the initial phase the convergence depends on the number of iterations; in the middle phase the rate depends on the condition number κ; and in the final phase one observes superconvergence. Usually the middle phase is longer if very small eigenvalues are present [134].

To give an impression of a typical eigenvalue distribution occurring in our context we choose as a first example a small problem with only four states and one output function. (Details of this example can be found in [27, Example 1].) Figure

27 displays the eigenvalues for a uniform mesh size of 2^{-9}; i.e., Λ includes all indices of scaling and wavelet functions up to scale 9. The arising condition number can be seen by considering the logarithmic scale, while gaps in the eigenvalue distribution are displayed in the linear scale. The large eigenvalues increase scale to scale; i.e., 2^i eigenvalues correspond to scale i of the wavelet discretization and have a magnitude of order 2^{2i}. This is, roughly speaking, due to the involved differential operator of order 2 in (204). On lower scales the model matrices are dominant in comparison to the differential operator. The scales cannot be distinguished. No isolated eigenvalues are present and the smallest eigenvalues are very close to zero. This leads, as expected, to a long middle phase in the convergence behavior.

Figure 27: Eigenvalue distribution.



Based on the knowledge of boundedness of the original operator from H^1 to $(H^1)'$ and its inverse, and using specific norm equivalences of wavelet representations, it is shown in [26] that scaling of the wavelet representations guarantees an upper limit for the number of iterations independently of ℓ. This scaling can also be viewed as symmetric, diagonal preconditioning with entries 2^{-j}. Instead of 2^{-j} an approximate Jacobi preconditioner $\boldsymbol{N}_{\Lambda^\ell}$ is used; i.e., the square root of an approximation of the diagonal of $\boldsymbol{X}_\Lambda$ is employed. The entries reflect the necessary scale-dependent scaling. In addition this preconditioner corresponds to normalizing the wavelet basis in the energy norm and usually gives better results. The arising condition numbers fulfill

$$\mathrm{cond}_2(\boldsymbol{N}_{\Lambda^\ell}\boldsymbol{X}_{\Lambda^\ell}\boldsymbol{N}_{\Lambda^\ell}) \leq c \tag{205}$$

independent of ℓ [26]. The scale-to-scale increase of the eigenvalues is eliminated. However, although the boundedness is guaranteed, the size of the constant c can be very large. In numerical experiments we see that the smallest eigenvalues are still close to zero and there is no gap in the lower spectrum (see, for example, Figure 28). Results for other diagonal preconditioners can be found in [27], where also various, more sophisticated preconditioners are discussed.

The smallest eigenvalues are due to inherent features of the estimation problem. The original problem is ill-posed and the measure of observability of the model is typically low. In addition, the stiffness of the involved ordinary differential equation plays a crucial role, since the initial values have to be estimated. To tackle the first problem an appropriate regularization parameter ε is chosen and the states are normalized by a rough guess of their maximum norm (see [27]). Nevertheless, this tuning of the problem formulation is still not sufficient to overcome the second and third problem-inherent features and to obtain a favorable eigenvalue distribution.

3 The Schur Complement Method to Eliminate Eigenvalues

3.1 Elimination of Eigenvalues

The idea of improving the eigenvalue distribution is based on the statements in the theorems given below. They confirm, roughly speaking, that if we choose a submatrix corresponding to the eigenspace of the smallest eigenvalues $0 < \lambda_1 \leq \cdots \leq \lambda_s$, then the Schur complement of this submatrix in the complete matrix has no eigenvalues smaller than λ_s. The small eigenvalues are eliminated. The theorems also include the symmetric indefinite case.

Theorem 3.1 *Suppose the matrix $\boldsymbol{X}$ is symmetric with eigenvalues $\lambda_1 \leq \cdots \leq \lambda_s < 0 < \lambda_{s+1} \leq \cdots \leq \lambda_n$ and eigenvectors $\boldsymbol{v}_i$ $(i = 1, \ldots, n)$. Furthermore, given a non-singular matrix $\boldsymbol{M}$ let us denote by ξ_i the eigenvalues of the symmetrically preconditioned matrix $\boldsymbol{M}\boldsymbol{X}\boldsymbol{M}^T$ with eigenvectors $\boldsymbol{w}_i$, where ξ_i is sorted in a non-decreasing order for $i = 1, \ldots, n$. Then, we have*

$$|\lambda_r| \min_{\boldsymbol{y}\in W_r} \frac{\boldsymbol{y}^T\boldsymbol{M}^T\boldsymbol{M}\boldsymbol{y}}{\boldsymbol{y}^T\boldsymbol{y}} \leq |\xi_r| \leq |\lambda_r| \max_{\boldsymbol{y}\in V_r} \frac{\boldsymbol{y}^T\boldsymbol{M}^T\boldsymbol{M}\boldsymbol{y}}{\boldsymbol{y}^T\boldsymbol{y}} \tag{206}$$

where

$$W_r = \begin{cases} \boldsymbol{M}^{-1} span\{\boldsymbol{w}_r, \ldots, \boldsymbol{w}_s\} & \text{if } r \leq s, \\ \boldsymbol{M}^{-1} span\{\boldsymbol{w}_{s+1}, \ldots, \boldsymbol{w}_r\} & \text{if } r > s, \end{cases} \tag{207}$$

$$V_r = \begin{cases} span\{\boldsymbol{v}_r, \ldots, \boldsymbol{v}_s\} & \text{if } r \leq s, \\ span\{\boldsymbol{v}_{s+1}, \ldots, \boldsymbol{v}_r\} & \text{if } r > s. \end{cases} \tag{208}$$

Proof: For $i = 1, \ldots, n-s$ the ith largest eigenvalue of $\boldsymbol{M}^{-T}\boldsymbol{X}^{-1}\boldsymbol{M}^{-1}$ is $1/\xi_{s+i}$. Hence, using the minimax characterization of the eigenvalues of symmetric matrices [89, 189] we have

$$\begin{aligned} 1/\xi_{s+i} &= \max_{dim U = i} \min_{\boldsymbol{y}\in U} \frac{\boldsymbol{y}^T\boldsymbol{M}^{-T}\boldsymbol{X}^{-1}\boldsymbol{M}^{-1}\boldsymbol{y}}{\boldsymbol{y}^T\boldsymbol{y}} \\ &\geq \min_{\boldsymbol{y}\in \boldsymbol{M}V_{s+i}} \left(\frac{\boldsymbol{y}^T\boldsymbol{M}^{-T}\boldsymbol{X}^{-1}\boldsymbol{M}^{-1}\boldsymbol{y}}{\boldsymbol{y}^T\boldsymbol{M}^{-T}\boldsymbol{M}^{-1}\boldsymbol{y}} \; \frac{\boldsymbol{y}^T\boldsymbol{M}^{-T}\boldsymbol{M}^{-1}\boldsymbol{y}}{\boldsymbol{y}^T\boldsymbol{y}} \right) \\ &\geq 1/\left\{ \lambda_{s+i} \max_{\boldsymbol{y}\in V_{s+i}} \frac{\boldsymbol{y}^T\boldsymbol{M}^T\boldsymbol{M}\boldsymbol{y}}{\boldsymbol{y}^T\boldsymbol{y}} \right\}, \end{aligned}$$

which gives the upper limit in (206). Exploiting the fact that $1/\lambda_{s+i}$ is the ith largest eigenvalue of $\boldsymbol{X}^{-1}$ for $i = 1, \ldots, n-s$ and using the space W_{s+i}, one can follow the above steps to obtain the lower limit. To obtain the estimates for the remaining negative eigenvalues we observe that $-1/\xi_{s+1-i}$ is the ith largest eigenvalue of $-\boldsymbol{M}^{-T}\boldsymbol{X}^{-1}\boldsymbol{M}^{-1}$ and use V_{s+1-i}, while $-1/\lambda_{s+1-i}$ is the ith largest eigenvalue of $-\boldsymbol{X}^{-1}$ and we can use $dim W_{s+1-i} = i$ for $i = 1, \ldots, s$. q.e.d.

A similar result can be obtained by directly considering the eigenvalues (and not their inverses) of $\boldsymbol{X}$ and $\boldsymbol{M}\boldsymbol{X}\boldsymbol{M}^T$. However, for elimination of the in modulo smallest eigenvalues the above result is more useful.

Before we present the next theorem let us define some notation. Suppose we partition the equation $\boldsymbol{X}\boldsymbol{d} = \boldsymbol{r}$ into

$$\begin{pmatrix} \boldsymbol{X}_{0,0} & \boldsymbol{X}_{0,J} \\ \boldsymbol{X}_{J,0} & \boldsymbol{X}_{J,J} \end{pmatrix} \begin{pmatrix} \boldsymbol{d}_0 \\ \boldsymbol{d}_J \end{pmatrix} = \begin{pmatrix} \boldsymbol{r}_0 \\ \boldsymbol{r}_J \end{pmatrix}, \tag{209}$$

where $\boldsymbol{X}_{0,0}$ is nonsingular, and let us denote the Schur complement by

$$\boldsymbol{\mathcal{S}} := \left(\boldsymbol{X}_{J,J} - \boldsymbol{X}_{J,0}\boldsymbol{X}_{0,0}^{-1}\boldsymbol{X}_{0,J}\right). \tag{210}$$

Then the solution of (209) is given by the solution of

$$\boldsymbol{\mathcal{S}}\boldsymbol{d}_J = \boldsymbol{r}_J - \boldsymbol{X}_{J,0}\boldsymbol{X}_{0,0}^{-1}\boldsymbol{r}_0 \tag{211}$$

followed by

$$\boldsymbol{d}_0 = \boldsymbol{X}_{0,0}^{-1}\left(\boldsymbol{r}_0 - \boldsymbol{X}_{0,J}\boldsymbol{d}_J\right). \tag{212}$$

Building the Schur complement $\boldsymbol{\mathcal{S}}$ corresponds to the application of symmetric preconditioning to (209). Namely, we obtain

$$\boldsymbol{M}\boldsymbol{X}\boldsymbol{M}^T = \begin{pmatrix} \boldsymbol{X}_{0,0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\mathcal{S}} \end{pmatrix} \quad \text{with} \quad \boldsymbol{M} := \begin{pmatrix} \boldsymbol{I} & \boldsymbol{0} \\ -\boldsymbol{X}_{J,0}\boldsymbol{X}_{0,0}^{-1} & \boldsymbol{I} \end{pmatrix}. \tag{213}$$

Hence the n eigenvalues of $\boldsymbol{M}\boldsymbol{X}\boldsymbol{M}^T$ are given by the n_0 eigenvalues of $\boldsymbol{X}_{0,0}$ and $n - n_0$ eigenvalues of $\boldsymbol{\mathcal{S}}$. The Sylvester law of inertia immediately yields the following proposition.

Proposition 3.1 *Let $\boldsymbol{X}$ have the inertia (s, z, p) and $\boldsymbol{X}_{0,0}$ the inertia (s_0, z_0, p_0). Then $\boldsymbol{\mathcal{S}}$ has the inertia $(s - s_0, z - z_0, p - p_0)$.*

A related result is given in [164]. There the Schur complement is formed with respect to a coarse finite element space to eliminate the negative eigenvalues for symmetric indefinite linear systems, which arise in elasticity problems. It is illustrated that this, also called the coarse space orthogonalization approach, is a key ingredient in obtaining satisfactory convergence for that class of problems. The Schur complement technique fits into the framework of the deflation technique. Deflation of a subspace

$\boldsymbol{Z}$ is obtained by the projection $\boldsymbol{P} = \boldsymbol{I} - \boldsymbol{X}\boldsymbol{Z}(\boldsymbol{Z}^T\boldsymbol{X}\boldsymbol{Z})^{-1}\boldsymbol{Z}^T$ of $\boldsymbol{X}\boldsymbol{d} = \boldsymbol{r}$. For positive definite matrices $\boldsymbol{X}$ it is known that deflation of an invariant subspace cancels the corresponding eigenvalues, leaving the rest of the spectrum untouched (see, for example, [80]). Hence, if $\boldsymbol{Z}$ is the invariant subspace to the ith smallest eigenvalues, then the effective condition number of $\boldsymbol{P}\boldsymbol{X}$ is given by λ_n/λ_{i+1}. However, for the Schur complement $\boldsymbol{Z} = (\boldsymbol{I}_m, \boldsymbol{0})^T$, which is in general not an invariant subspace. In the following theorem we assume the weaker condition that the subspace $\boldsymbol{Z}$ contains the in modulo smallest eigenvalues and quantify for this case the location of the eigenvalues of $\boldsymbol{\mathcal{S}}$.

Theorem 3.2 *Let $\boldsymbol{X}$ have eigenvectors of the form $\boldsymbol{v}_l = (\boldsymbol{v}_{l,0}^T, \boldsymbol{0}^T)^T$ for $l = s-i', \dots, s+i$. Moreover, let the corresponding eigenvalues λ_l also be the in modulo smallest eigenvalues of $\boldsymbol{X}_{0,0}$. Then $\boldsymbol{\mathcal{S}}$ has eigenvalues ζ_j with*

$$\zeta_1 \leq \cdots \leq \zeta_{s-s_0} \leq \lambda_{s-i'} \leq \lambda_{s+i} \leq \zeta_{s-s_0+1} \leq \cdots \leq \zeta_{n-n_0}. \tag{214}$$

Proof: To prove the statement we show that $\xi_l = \lambda_l$. We use (206) and consider the image of the relevant eigenspaces of $\boldsymbol{X}$ and $\boldsymbol{M}\boldsymbol{X}\boldsymbol{M}^T$ with respect to $\boldsymbol{M}$.

Given the special structure of the eigenvectors of $\boldsymbol{X}$ for $l \in \{s-i', \dots, s+i\}$, we have $\boldsymbol{X}_{0,0}\boldsymbol{v}_{l,0} = \lambda_l \boldsymbol{v}_{l,0}$ and, in addition, $\boldsymbol{X}_{J,0}\boldsymbol{v}_{l,0} = \boldsymbol{0}$. It follows that $\boldsymbol{M}\boldsymbol{v}_l = \boldsymbol{v}_l$. Hence, $\boldsymbol{M}\boldsymbol{y} = \boldsymbol{y}$ for all $\boldsymbol{y} \in V_l$ and we deduce that $|\xi_l| \leq |\lambda_l|$.

For $\xi_r = \lambda_l$ we have $\boldsymbol{w}_r = \boldsymbol{v}_l$ and consequently, $\boldsymbol{M}\boldsymbol{w}_r = \boldsymbol{w}_r$. The eigenvector corresponding to $\xi_r = \zeta_j$ is given by $\boldsymbol{w}_r = (\boldsymbol{0}^T, \boldsymbol{w}_{r,J}^T)^T$ and we can deduce that $\boldsymbol{M}\boldsymbol{w}_r = \boldsymbol{w}_r$. Furthermore, if we denote by σ_j the remaining eigenvalues of $\boldsymbol{X}_{0,0}$, then $\boldsymbol{w}_r = (\boldsymbol{w}_{r,0}^T, \boldsymbol{0}^T)^T$ and $\boldsymbol{X}_{J,0}\boldsymbol{w}_{r,0} \neq \boldsymbol{0}$ hold for $\xi_r = \sigma_j$. This leads to $\boldsymbol{M}\boldsymbol{w}_r \neq \boldsymbol{w}_r$. However, in accordance to the assumption, we have $\sigma_j \leq \lambda_{s-i'}$ or $\lambda_{s+i} \geq \sigma_j$, which implies $r < s-i'$ or $r > s+i$ for $\xi_r = \sigma_j$. Thus $\boldsymbol{w}_r \notin \{\boldsymbol{w}_{s-i'}, \dots, \boldsymbol{w}_{s+i}\}$ and we conclude $\boldsymbol{M}\boldsymbol{y} = \boldsymbol{y}$ for all $\boldsymbol{y} \in W_l$, which yields $|\lambda_l| \leq |\xi_l|$. q.e.d.

Using $\lambda_{max} = \max_{\boldsymbol{y}\neq 0}(\boldsymbol{y}^T\boldsymbol{X}\boldsymbol{y})/(\boldsymbol{y}^T\boldsymbol{y})$ and the subspace $\boldsymbol{y}_0 = -\boldsymbol{X}_{0,0}^{-1}\boldsymbol{X}_{0,J}\boldsymbol{y}_J$, we find that the following corollary holds for positive definite systems.

Corollary 3.1 *Let $\boldsymbol{X}$ be symmetric positive definite and in agreement with the assumption in Theorem 3.2. Then*

$$\lambda_i \leq \zeta_1 \leq \zeta_{n-n_0} \leq \lambda_n, \tag{215}$$

giving a reduced condition number $cond(\boldsymbol{\mathcal{S}}) \leq \lambda_n/\lambda_i$.

Obviously we can also apply the above ideas to eliminate the in modulo largest eigenvalues.

3.2 The Choice of the Subspace

Based on Theorem 3.2, efficient preconditioning is built on the following consideration: If we would find a permutation of $\boldsymbol{X}$ such that $\boldsymbol{X}_{0,0}$ inhabits the ill-conditioning of $\boldsymbol{X}$, i.e., has the smallest eigenvalues of $\boldsymbol{X}$, and if the size of $\boldsymbol{X}_{0,0}$ is very small compared with $\boldsymbol{\mathcal{S}}$, then the Schur complement $\boldsymbol{\mathcal{S}}$ is still large in size but is so well conditioned that the application of an iterative solver is very efficient. In general, it is difficult to determine a subspace and with it a suitable permutation of $\boldsymbol{X}$ such that the assumptions of the theorem are fulfilled. It may even happen that there is no suitable reordering apart from the trivial choice $\boldsymbol{X}_{0,0} = \boldsymbol{X}$. Here, the basis in which $\boldsymbol{X}$ represents the corresponding linear map is crucial. In our application the map is represented in a wavelet basis. The arising matrices $\boldsymbol{X}_{\Lambda^\ell}$ are discretization matrices and we can associate with the submatrix $\boldsymbol{X}_{0,0}$ a discretization space V_0 and, respectively, an index set Λ_0. Several questions arise: Is it necessary to fulfill $\boldsymbol{v}_l \in V_0$ exactly? Does a set Λ_0 exist that is coarse? How can we find such a set? Of course there are many more open questions, for example: How many eigenvalues would we like to eliminate and how do we invert the ill-conditioned but small matrix $\boldsymbol{X}_{0,0}$? From the reflections below and the numerical analysis in section 4 we see that employing the Schur complement method yields encouraging answers to the first three questions.

For the choice of a suitable subspace we keep to the spirit of wavelet discretization. Namely, the advantage of using wavelets is that the (infinite dimensional) wavelet discretization is a representation of the original problem. The system of functional equations is transformed into a system of discrete equations. This discrete system is then projected on a hierarchy of subspaces to find approximate solutions. Hence, corresponding to the smallest eigenvalues of the infinite dimensional problem we choose a subspace determined by an index set Λ_0, which shall capture good approximations of the eigenvectors, or, equivalently, an approximation space $\{\psi_{j,k} | \{j,k\} \in \Lambda_0\}$ for the eigenfunctions. Then, this subspace shall also serve as a subspace for the Schur complement method for all principal submatrices $\boldsymbol{X}_{\Lambda^\ell}$, $\ell = 0, 1, \ldots$. Here, we take into account the fact that only the large eigenvalues increase with the scale, while the small eigenvalues stay roughly the same. Exploiting the Riesz basis property and with it the good compression property of wavelets, one expects to find a very coarse space, respectively, a set Λ_0 with few indices, which contains a good approximation of the relevant eigenspace.

Now let us consider the original problem. For $\boldsymbol{W} = \boldsymbol{I}$ the first order conditions (197)–(198) reduce to $\langle \bar{\boldsymbol{\mathcal{S}}} \boldsymbol{x}, \xi \rangle = \langle \boldsymbol{z}, \boldsymbol{C}\xi \rangle + \varepsilon \langle \boldsymbol{u}, \dot{\xi} - \boldsymbol{A}\xi \rangle$ with $\langle \bar{\boldsymbol{\mathcal{S}}} \boldsymbol{x}, \xi \rangle := \varepsilon \langle \dot{\boldsymbol{x}} - \boldsymbol{A}\boldsymbol{x}, \dot{\xi} - \boldsymbol{A}\xi \rangle + \langle \boldsymbol{C}\boldsymbol{x}, \boldsymbol{C}\xi \rangle$. Using the connection with the Sturm–Liouville eigenvalue problem one can show the following (details can be found in [33]).

Proposition 3.2 *If we consider the estimation problem (190)–(192) with one-dimensional constraints only, then the operator $\bar{\boldsymbol{\mathcal{S}}} : H^1 \to (H^1)'$ has infinite, countable many eigenvalues λ_i, where λ_i behaves like i^2. The smallest eigenvalue is $\lambda_0 = C^2$ with eigenfunction* $\exp(At)$.

Hence, Proposition 3.2 motivates us to employ an adapted index set Λ_0, which provides for large A a good wavelet approximation of $exp(At)$. This corresponds to the choice of a nonuniform grid with grid points close to the origin.

3.3 Preconditioning the Schur Complement

Before we present the numerical results, we would like to bring to the reader's attention the fact that symmetric preconditioning with diagonal blocks of sizes as in $\boldsymbol{M}$ and constructing the Schur complement can be interchanged. This follows since

$$\underbrace{\begin{pmatrix} \boldsymbol{N}_0 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{N}_J \end{pmatrix}}_{:=\boldsymbol{N}} \begin{pmatrix} \boldsymbol{I} & \boldsymbol{0} \\ -\boldsymbol{X}_{J,0}\boldsymbol{X}_{0,0}^{-1} & \boldsymbol{I} \end{pmatrix} = \underbrace{\begin{pmatrix} \boldsymbol{I} & \boldsymbol{0} \\ -\bar{\boldsymbol{X}}_{J,0}\bar{\boldsymbol{X}}_{0,0}^{-1} & \boldsymbol{I} \end{pmatrix}}_{:=\overline{\boldsymbol{M}}} \begin{pmatrix} \boldsymbol{N}_0 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{N}_J \end{pmatrix},$$

where $\bar{\boldsymbol{X}}_{J,0} = \boldsymbol{N}_J\boldsymbol{X}_{J,0}\boldsymbol{N}_0^T$ and $\bar{\boldsymbol{X}}_{0,0} = \boldsymbol{N}_0\boldsymbol{X}_{0,0}\boldsymbol{N}_0^T$ are the matrices of the preconditioned matrix of $\boldsymbol{N}\boldsymbol{X}\boldsymbol{N}^T$. This ensures that the preconditioning presented in subsection 2.4 also provides an efficient preconditioner for the Schur complement.

Proposition 3.3 *Let $\boldsymbol{N}_\Lambda$ be given as in (205). Then, the condition numbers of the preconditioned Schur complements*

$$\kappa^\ell := cond_2(\boldsymbol{N}_{\Lambda^\ell_{J,J}}\boldsymbol{\mathcal{S}}_{\Lambda^\ell}\boldsymbol{N}_{\Lambda^\ell_{J,J}}) \leq c \tag{216}$$

are bounded independently of the refinement step ℓ.

Proof: Since for symmetric positive definite matrices $\boldsymbol{X}$ the estimations $cond(\boldsymbol{\mathcal{S}}) \leq cond(\boldsymbol{X})$ and $cond(\boldsymbol{X}_{0,0}) \leq cond(\boldsymbol{X})$ hold (see [9]), we have $cond(\boldsymbol{M}\boldsymbol{X}\boldsymbol{M}^T) \leq cond(\boldsymbol{X})$. Hence, using (205) we have

$$\begin{aligned} cond(\boldsymbol{N}_{\Lambda^\ell_{J,J}}\boldsymbol{\mathcal{S}}_{\Lambda^\ell}\boldsymbol{N}_{\Lambda^\ell_{J,J}}) &\leq cond(\boldsymbol{N}_{\Lambda^\ell}\boldsymbol{M}\boldsymbol{X}_{\Lambda^\ell}\boldsymbol{M}^T\boldsymbol{N}_{\Lambda^\ell}) \\ &= cond(\overline{\boldsymbol{M}}\boldsymbol{N}_{\Lambda^\ell}\boldsymbol{X}_{\Lambda^\ell}\boldsymbol{N}_{\Lambda^\ell}\overline{\boldsymbol{M}}^T) \\ &\leq cond(\boldsymbol{N}_{\Lambda^\ell}\boldsymbol{X}_{\Lambda^\ell}\boldsymbol{N}_{\Lambda^\ell}) \leq c \end{aligned}$$

independent of the refinement step ℓ. q.e.d.

In practice we employ instead of $\boldsymbol{N}_{\Lambda^\ell_{J,J}}$ an approximation $\boldsymbol{B}^\ell$ of the square root of the diagonal of the Schur complement.This matrix exhibits the necessary scale-dependent scaling as explained in subsection 2.4 and corresponds to normalizing the wavelet basis in the energy norm with respect to $\boldsymbol{\mathcal{S}}_{\Lambda^\ell}$. Numerical investigations confirmed that $\boldsymbol{B}^\ell$ clearly yields the best results compared with various diagonal preconditioners having entries with the relevant scale dependency, such as diagonal entries 2^{-j}, the diagonal of $\boldsymbol{X}_{J,J}$, and the matrix $\boldsymbol{N}_{\Lambda^\ell_{J,J}}$. A factor of 100 between the condition numbers is not unusual, although though all preconditioners provide asymptotically a scale-independent behavior of the condition numbers.

4 Numerical Investigations

4.1 The Algorithm

In summary, we obtain the following sketched algorithm. Here the upper index ℓ denotes the ℓth nested iteration while the lower indices 0 and J correspond to the splitting of the matrix $\boldsymbol{X}^\ell = \boldsymbol{X}_{\Lambda^\ell}$ for the Schur complement. Due to the real-time restriction the nested iteration is limited by time. However, this stopping criteria can easily be replaced, for example, by residual-based criteria using the residual $\boldsymbol{X}^{\ell+1}\boldsymbol{d}^\ell - \boldsymbol{r}^{\ell+1}$ on the next refinement. We would also like to remind the reader that the matrix $\boldsymbol{X}^\ell$ is a result of a reduction step (see (203)). Hence the matrix $\boldsymbol{X}^\ell$ is not only an extension of $\boldsymbol{X}^{\ell-1}$. However, we omit the details for easier readability.

initialize Λ_0
set $\ell = 0$

- *define* $\Lambda^\ell := \Lambda_0$
- *determine* $\boldsymbol{r}^\ell$ *and* $\boldsymbol{X}^\ell$
- *solve with a direct method* $\boldsymbol{X}^\ell \boldsymbol{d}^\ell = \boldsymbol{r}^\ell$

while time allows set $\ell = \ell + 1$ *and:*

- *determine adaptively* Λ^ℓ
- *set* $\Lambda_0^\ell := \Lambda_0$ *and* $\Lambda_J^\ell := \Lambda^\ell \setminus \Lambda_0$
- *update* $\boldsymbol{X}^0$ *to* $\boldsymbol{X}_{0,0}^\ell$
- *assemble wavelet matrices* $\langle\langle \boldsymbol{\Psi}_{\Lambda_0}, \boldsymbol{\Psi}_{\Lambda_J^\ell} \rangle\rangle$, *etc. which are involved in* $\boldsymbol{X}_{J,0}^\ell$ *and* $\boldsymbol{X}_{J,J}^\ell$ *to realize their applications to a vector*
- *determine the preconditioner* $\boldsymbol{B}^\ell$ *using only the diagonals of all involved matrices*
- *update* $\boldsymbol{r}^0$ *to* $\boldsymbol{r}_0^\ell$ *and determine* $\boldsymbol{r}_J^\ell$
- *determine* $\boldsymbol{r}_{\mathcal{S}}^\ell = \boldsymbol{r}_J^\ell - \boldsymbol{X}_{J,0}^\ell (\boldsymbol{X}_{0,0}^\ell)^{-1} \boldsymbol{r}_0^\ell$ *using a direct solver for* $(\boldsymbol{X}_{0,0}^\ell)^{-1}\boldsymbol{r}_0^\ell$ *and applying* $\boldsymbol{X}_{J,0}^\ell$
- *extend* $\boldsymbol{d}_J^{\ell-1}$ *with zeros to define the initial guess* $\boldsymbol{d}_J^\ell$
- *apply the pcg method to solve* $\boldsymbol{\mathcal{S}}^\ell \boldsymbol{d}_J^\ell = \boldsymbol{r}_{\mathcal{S}}^\ell$ *using the initial guess* $\boldsymbol{d}_J^\ell$ *and the preconditioner* $\boldsymbol{B}^\ell$ *until a given tolerance is reached or the number of maximal iterations is exceeded (the application of* $\boldsymbol{\mathcal{S}}^\ell$ *to a vector is realized by applying* $\boldsymbol{X}_{J,0}^\ell$, $\boldsymbol{X}_{J,J}^\ell$, *and a direct solver for* $(\boldsymbol{X}_{0,0}^\ell)^{-1}$*)*
- *solve* $\boldsymbol{X}_{0,0}^\ell \boldsymbol{d}_0^\ell = \boldsymbol{r}_0^\ell - \boldsymbol{X}_{0,J}^\ell \boldsymbol{d}_J^\ell$
- *set* $\boldsymbol{d}^\ell = ((\boldsymbol{d}_0^\ell)^T, (\boldsymbol{d}_J^\ell)^T)^T$

end

The complexity of one nested iteration step ℓ is therefore given by the following operations: the setup of $\boldsymbol{X}_{0,0}^{\ell}$, which results in $O(n_x^3(\#\Lambda_0)^2\#\Lambda_J^{\ell}) + O(n_x^3(\#\Lambda_0)^3)$ operations; the setup of $(\boldsymbol{X}_{0,0}^{\ell})^{-1}$ (for example with Cholesky decomposition) with $O(n_x^3(\#\Lambda_0)^3)$ operations; the determination of $\boldsymbol{B}^{\ell}$ with $O(n_x\#\Lambda_J^{\ell})$ operations; and then for each pcg iteration steps the applications of $\boldsymbol{X}_{J,0}^{\ell}$, $\boldsymbol{X}_{J,J}^{\ell}$, and $(\boldsymbol{X}_{0,0}^{\ell})^{-1}$, which result in $O(n_x^2\#\Lambda_J^{\ell} log(\#\Lambda_J^{\ell}))+O(n_x^2(\#\Lambda_0)^2)$ operations. Considering $\ell \to \infty$ one can think of applying the algorithm proposed in [45] for the matrix-vector multiplication involving $\boldsymbol{X}_{J,0}^{\ell}$, $\boldsymbol{X}_{J,J}^{\ell}$. This may lead to a complexity without $log(\#\Lambda_J^{\ell})$. However, a detailed study of the applicability is not yet completed. Moreover, one has to keep in mind that in practice n_x can be very large, and then the real-time limitation restricts $\#\Lambda^{\ell}$. The number of necessary pcg iterations in each nested iteration step ℓ is bounded independently of ℓ due to (216).

4.2 Eigenvalue Distribution and Condition Numbers

Let us first illustrate the effects of the Schur complement method and of the diagonal preconditioning with our example from subsection 2.4. As Λ_0 we choose all wavelets up to scale four and then choose up to scale eight the two wavelets at the boundary $t = 0$, i.e., $\Lambda_0 = \{(0,0), \ldots, (3,7); (i,0), (i,1) \text{ for } i = 4, \ldots, 7\}$. The condition numbers behave as listed in Table 18. As can be seen, wavelet preconditioning alone would not be sufficient, while the combination with the Schur complement technique reduces the condition number by a factor of 1.e+6.

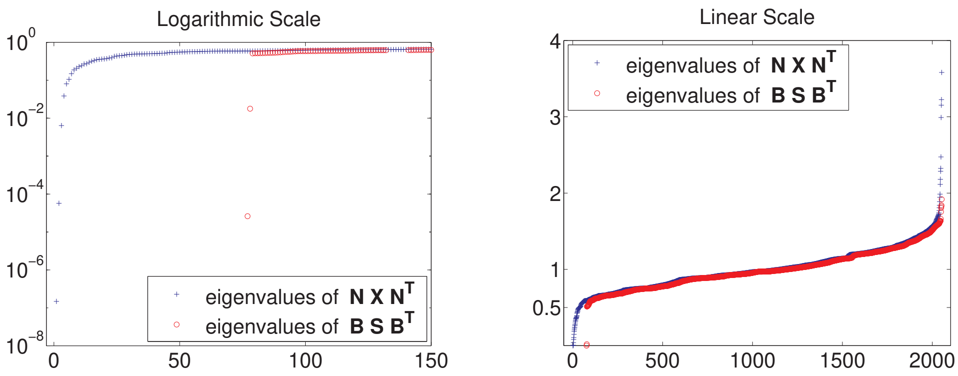
Table 18: Condition numbers on scale 9 ($h = 2^{-9}$).

	$\boldsymbol{X}$	$\boldsymbol{N}\boldsymbol{X}\boldsymbol{N}^T$	$\boldsymbol{\mathcal{S}}$	$\boldsymbol{B}\boldsymbol{\mathcal{S}}\boldsymbol{B}^T$
cond.	2.59e+10	2.42e+07	2.28e+06	7.36e+04

Figure 28 displays the eigenvalue distributions of $\boldsymbol{N}\boldsymbol{X}\boldsymbol{N}^T$ and $\boldsymbol{B}\boldsymbol{\mathcal{S}}\boldsymbol{B}^T$ on scale 9. Since the Schur complement $\boldsymbol{\mathcal{S}}$ has less eigenvalues we associated the indices of the eigenvalues with the employed index set Λ, respectively, with Λ/Λ_0, and listed them along the horizontal axis. We see that the scale-to-scale increase is extinguished. In addition, the Schur complement technique eliminates the smallest and the eleven largest eigenvalues of $\boldsymbol{N}\boldsymbol{X}\boldsymbol{N}^T$. Despite that the preconditioned Schur complement $\boldsymbol{B}\boldsymbol{\mathcal{S}}\boldsymbol{B}^T$ still has two small eigenvalues ($2.6e-5$, 0.02), we see from the linear scaled plot that a large gap arises to the third smallest eigenvalue (0.51). This has, as mentioned in subsection 2.4, a large impact on the convergence behavior of the cg-method, and the effect can be seen on the iteration numbers listed in the next subsection.

As a second test problem we consider an example which inherits two main features occurring in process monitoring. First, the involved differential equations are stiff and, second, the model is nearly not observable. It is based on a simple model of a chemical process which produces ethylene glycol from ethylene oxide and glycol. The nonlinear model lumps a number of related states and outputs. Further

Figure 28: Eigenvalue distribution of $\boldsymbol{N X N}^T$ and $\boldsymbol{B \mathcal{S} B}^T$.



simplification is accomplished by linearization at some stationary operating point and by subsequent elimination of the algebraic state variables to result in a linear model with 12 states and 5 measurements. Details on the model can be found in [27] and references therein. To ensure that the relative errors in the states are equally weighted, the model matrices are scaled such that the components of the nominal solution are roughly of unit size in L_∞. For brevity we do not give the numerical values of the system matrices. However, the eigenvalues of the model matrix $\boldsymbol{A}$ provide insight into system behavior and the values are given by $\{-189.90, -155.14, -3.78, -2.34, -1.98, -1.94, -1.15, -1.08 \pm i\, 0.12, -0.89, -6.17e{-}6, 0\}$. The state x_2 corresponds to the eigenvalue 0 and depends completely on the initial value, which has to be estimated. Moreover, this state has a very low observability measure.

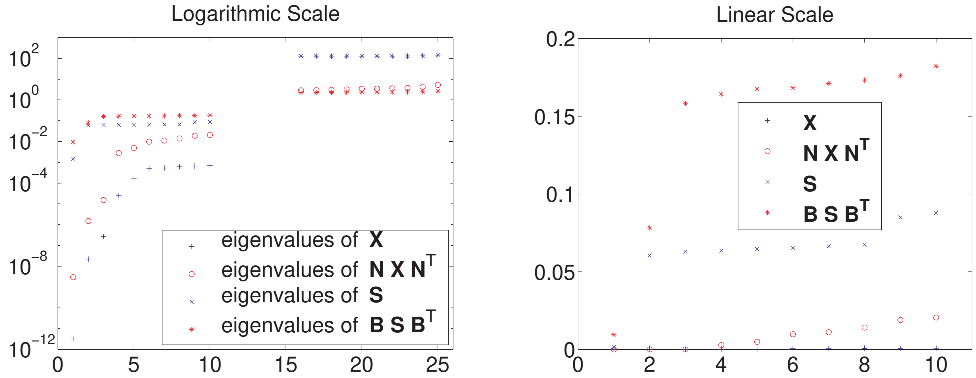
Given the original problem formulation, we were faced with condition numbers around **1e+20** already on scale 7. Scaling of the states to roughly unit size in L_∞ and regularization (with $\varepsilon = 2e - 4$) lead to condition numbers around 1e+13 on scale 9. However, only the application of the Schur complement technique reduced the condition number to a reasonable size, namely 280, as can be seen in Table 19 for scale 9. We employed an index set with only 13 indices for each state, to be more precise, $\Lambda_0 = \{(0,0); \ldots; (2,3); (3,0); (4,0); (5,0); (6,0)\}$.

Table 19: Condition numbers on scale 9 ($h = 2^{-9}$).

	$\boldsymbol{X}$	$\boldsymbol{N X N}^T$	$\boldsymbol{\mathcal{S}}$	$\boldsymbol{B \mathcal{S} B}^T$
cond.	**4.52e+13**	1.79e+09	9.51e+04	**2.80e+02**

Figure 29 displays on the left the ten smallest and largest eigenvalues in logarithmic scale while on the right displays only the ten smallest eigenvalues in linear scale. The largest eigenvalues of $\boldsymbol{X}$ and $\boldsymbol{\mathcal{S}}$ as well as of $\boldsymbol{N X N}^T$ and $\boldsymbol{B \mathcal{S} B}^T$ are (nearly) the same. However, the factor between the smallest eigenvalue of $\boldsymbol{X}$ and $\boldsymbol{B \mathcal{S} B}^T$ is 1e+10. Moreover, as can be seen in the plot in linear scale, the Schur complement technique gives rise to large gaps between the smallest eigenvalues, much in favor of the cg-method.

Figure 29: The ten smallest and largest eigenvalues.



4.3 Iteration Numbers and L_2-Errors

The iteration numbers are reduced more drastically than the condition number due to the arising gaps in the eigenvalue distribution. For instance, for our first example they drop from 973 to 22 for the applied pcg-method given the tolerance $tol = 1.e-6$ for the relative residual (see Table 20). Moreover, a comparable error in the case of employing $\boldsymbol{X}$ is only reached with $tol = 1.e-10$ and requires 1650 iterations. Obviously, the presented wavelet-Schur-complement preconditioning yields not only lower iteration numbers but also a considerably decreased error. This effect is observed in general.

Table 20: pcg-method applied to $\boldsymbol{X}$ and $\boldsymbol{B\mathcal{S}B}^T$ on scale 9.

	tol.	iter.	$\|\|e_1\|\|_{L_2}$	$\|\|e_2\|\|_{L_2}$	$\|\|e_3\|\|_{L_2}$	$\|\|e_4\|\|_{L_2}$
$\boldsymbol{X}$	1.e-6	973	1.59e-1	6.56e-2	2.63e-2	3.61e-3
$\boldsymbol{X}$	1.e-10	1650	4.36e-5	1.22e-5	6.33e-6	3.45e-6
$\boldsymbol{B\mathcal{S}B}^T$	1.e-6	22	4.77e-5	1.53e-5	5.30e-6	2.91e-6

For our second example (where we started with a condition number of about 1e+20), we obtain iteration numbers around 23, independent of the chosen degree of refinement, which can be seen in Table 21. As tolerance for the relative residuals we use $tol = 1e-04$.

Table 21: Various scales for $\boldsymbol{B\mathcal{S}B}^T$.

Scale	7	8	9	10
Number of unknowns	3729	7441	14865	29713
Iterations	22	25	23	22

The resulting L_2-norms of the errors between the obtained approximations and the exact state functions are listed for the first five states in Table 22, while increasing the employed index set Λ^ℓ scalewise. With the presented preconditioning technique the wavelet-Galerkin discretization yields reasonable results even for the

Table 22: L_2-norms of the errors in the states using $\boldsymbol{BSB}^T$.

Scale	7	8	9	10
x_1	7.7178e-04	1.7699e-04	3.7967e-05	6.5373e-06
x_2	6.1436e+00	1.1187e+00	3.3672e-01	7.2894e-02
x_3	7.7386e-03	1.7650e-03	3.6426e-04	5.8558e-05
x_4	9.4941e-03	2.3140e-03	5.6531e-04	1.3459e-04
x_5	3.2000e-01	8.1444e-02	2.0382e-02	5.3118e-03

hardly observable state x_2. The rate of convergence behaves as expected, i.e., for step-sizes $h_i = 2^i$ we observe the quotient $4 = (h_i/h_{i+1})^2$ between the errors of succeeding scales. A tighter choice for the tolerance *tol* does not lead to higher accuracy. This confirms that $tol = 1e-04$ is sufficient. In contrast the SQOPT algorithm [87], designed in particular for quadratic optimization problems, stops after 500 iterations without result on scales larger than 6.

For the discussion of the adaptive refinement of Λ^ℓ we refer to other literature [23, 28]. We would like to mention at this point only that an appropriate choice of Λ^ℓ with much less indices would also guarantee similar L_2-errors. However, we face the same problems concerning the eigenvalue distribution: small eigenvalues are close to zero and large eigenvalues increase with the used scales. By employing the above preconditioning techniques, both features can be controlled. The required iteration numbers remain around 24, independent of the refinement level, and we obtain the maximal convergence order given by the discretization spaces. The SQOPT algorithm failed for the tested adaptive choices of Λ^ℓ with $\#\Lambda^\ell > 100$.

In summary we can say that for all examples we tested, the Schur complement technique combined with wavelet preconditioning provides small, scale-independent iteration numbers and in addition an improved estimation quality with lower necessary tolerances in the residuals.

4.4 Influence of the choice of Λ_0

To gain insight into the influence of the choice of Λ_0, we compared several choices of Λ_0 for our second example with respect to resulting eigenvalue distributions, condition numbers, required tolerances, iteration numbers, as well as final cpu times according to $\boldsymbol{BSB}^T$.

First we calculated the 10 smallest eigenvalues and eigenvectors of the preconditioned matrix $\boldsymbol{NXN}^T$ on scale 9. Then, we approximated the eigenvectors of the 3 or, respectively, 10 smallest eigenvalues based on the norm equivalences for wavelets. We either used the N in modulo largest wavelet coefficients and, hence, employed as a measure for the distance to the relevant eigenspace the sum over the approximations to all considered eigenvectors, or restricted the maximal approximation error δ for each eigenvector. In the first case we considered various numbers N of largest coefficients; in the other case we varied the allowed approximation errors δ. We also considered the choice of Λ_0, as motivated in subsection 3.2, by varying the number of the wavelet functions employed at the beginning of the interval. In particular, we used the choice $\Lambda_0^* := \{(0,0); \ldots; (2,3); (3,0); (4,0); (5,0); (6,0)\}$, which was presented in subsection 4.2. On top of these variations, which have no requirement on the arising structure of index distributions, we either considered the additional requirement of a so-called tree structure of Λ_0, or assumed that the three coarsest levels are contained in the index set, i.e., $\{(0,0); \ldots; (2,3)\} \subseteq \Lambda_0$. The number of employed wavelets varied between 9 and 231 and the approximation error was, in the best case, roughly $9.2e-3$. In the figures below the variation of Λ_0 is sorted along the x-axis, and for each requirement on the index structure a new line or an additional figure is plotted. Numbers 1–19 correspond to the variation of N and δ, while 20–22 belong to the variation at the beginning of the interval. Number 22 corresponds to Λ_0^*.

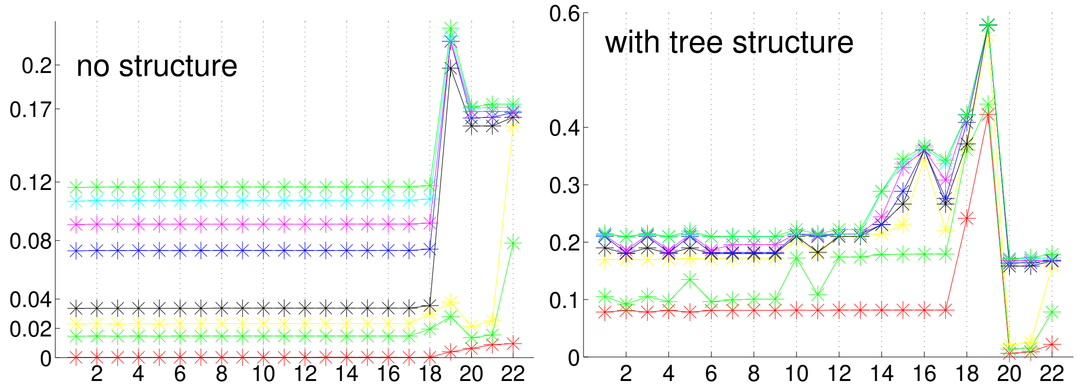
4.4.1 Eigenvalue Distribution

The variance between the largest eigenvalues arising with the different choices of Λ_0 is, as expected, negligible. The largest eigenvalues stay between 3.585 and 1.463, while the largest eigenvalue of $\boldsymbol{NXN}^T$ is 5.321. Our application of the Schur complement technique aims to influence the smallest eigenvalues. Here large differences arise. While the smallest eigenvalue of $\boldsymbol{NXN}^T$ is $2.966e-9$, for $\boldsymbol{B\mathcal{S}B}^T$ they vary between $2.488e-4$ and $4.224e-1$. Clearly, for all choices $\boldsymbol{B\mathcal{S}B}^T$ has condition numbers which are reduced by at least a factor of $1e-5$. In the best case the condition number reduced to 3.46, although here we employed an index set with $\#\Lambda_0 = 231$ per state (number 19). (The complete index set on scale 9 inhabits 513 indices.) The condition number 29.08 is reached with $\#\Lambda_0 = 42$ and 279.47 with Λ_0^*, which has $\#\Lambda_0^* = 13$.

Using the various approximations of the eigenvectors the minimal eigenvalues are roughly the same as long as we do not incorporate drastically more indices in the set Λ_0 (as for numbers 18 and 19 in the plots). This indicates that the way of measuring the distance to the relevant eigenspace has only a small, if any, influence. Moreover, it is surprising that three eigenvalues were extinguished for any considered Λ_0, while the partly desired elimination of ten eigenvalues was only attained after the additional requirement of a tree structure for the indices. Hence, it seems to be difficult to eliminate the eigenvalues based only on approximating the corresponding eigenvectors. A significant difference can be found for the choices of wavelets at the beginning of the interval. Such behavior is observed in all numerical experiments.

As mentioned, also gaps in the eigenvalue distribution strongly influence the efficiency of the cg-method. Figure 30 displays the distribution of the eight smallest eigenvalues of $\boldsymbol{B\mathcal{S}B}^T$. Obviously without structure Λ_0^* (number 22) has the most favorable distribution, namely large gaps between the smallest, second smallest, and third smallest eigenvalue. Only with a required tree structure are the gaps between the eigenvalues improved to the extent that 1–19 beat 22. However, their index sets include Λ_0 of 22, which is Λ_0^* extended to tree structure. Of course not only the eigenvalue distribution but also the number of employed wavelets play a significant role in the efficiency, as can be seen later on.

Figure 30: The eight smallest eigenvalues of $\boldsymbol{B}\boldsymbol{\mathcal{S}}\boldsymbol{B}^T$.



4.4.2 Iteration numbers

One important stopping criteria of an iterative method is the tolerance for the (relative) residual. It turned out that, in general, $tol = 1.e-4$ is not sufficient to guarantee the best attainable approximation error given by the discretization error of the optimization problem. In all cases a tolerance for the relative residual of $tol = 1.e-6$ led to the same approximation errors fulfilling the desired requirements. Hence the results in the next subsections are based on the use of $tol = 1.e-6$.

As the eigenvalue distributions indicated already, Λ_0^* provides the lowest iteration numbers over the scales, neither assuming index structure nor incorporating the coarsest three scales (34, 31, and 30 iterations on scale 8, 9, and 10, respectively). Requiring tree structure, 22 needs 31, 27, and 26 iterations, while 15 with $\#\Lambda_0 = 96$ needs 16, 14, and 16 iterations and 18 with $\#\Lambda_0 = 168$, which can only start on scale 9, needs 10 and 11 iterations. Figure 31 presents graphically the average of the necessary iteration numbers over the scales.

Figure 31: Averaged iteration numbers of the pcg-method applied to $\boldsymbol{B}\boldsymbol{\mathcal{S}}\boldsymbol{B}^T$.

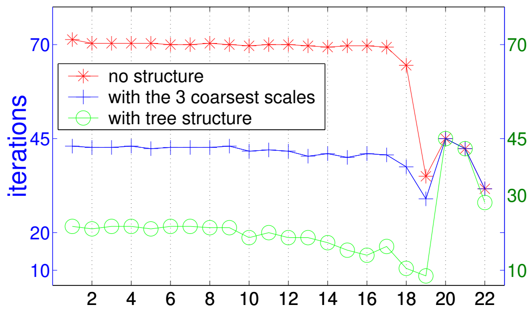
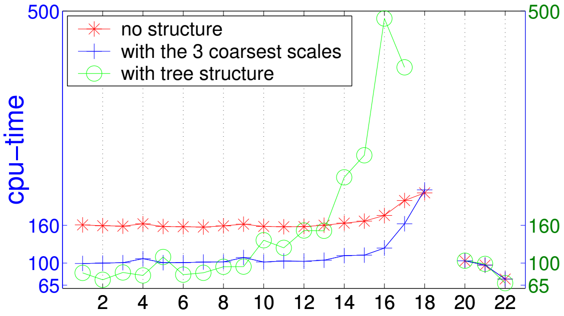


Figure 32: Cpu time of the nested iteration applied to $\boldsymbol{B}\boldsymbol{\mathcal{S}}\boldsymbol{B}^T$.



4.4.3 Cpu Time

Figure 32 displays the required cpu times. For scaling reasons only values below 500s are plotted. Here, the computation time is measured, which is required for the algorithm presented in subsection 4.1, with $\Lambda^1, \Lambda^2, \Lambda^3$ chosen as the complete scales 8, 9, 10, respectively. Only for 18 and 19 did we start at the minimal scale 9, such that the computation times are not really comparable. Nevertheless, it shows that 18 and 19 already require for two scales more computation time than the others for three scales. The computations are executed on an IntelXeon processor with 4GB and 700MHz. Moreover, we would like to emphasize that the code used is not yet time optimal and is used for academic purposes only. However, since the experiments are all run with the same code, the comparison of the times between the various Λ_0 is still valid.

We can clearly observe the interaction between the improved eigenvalue distribution and the number of employed indices with respect to the efficiency measured by the cpu time. Less numbers of indices can reduce the influence of larger condition numbers and vice versa. Due to the best eigenvalue distributions and the very low number of indices, Λ_0^* is the fastest in the case of either no requirement on the index structure or of incorporating the three coarsest levels. This choice leads to 74.69s, while 1–16 need approximately 160s in the first case and approximately 100s in the second case. In case of the tree structure it depends very much on the number of employed indices. Choice 22 is still the most efficient with 68.26s despite its condition number, which is at least 20 times as large as for choice 18 and 19. The originally chosen set Λ_0^* is only 6s slower.

In summary, we have to say that a general guideline for the choice of Λ_0 in order to obtain the optimal efficiency cannot be provided so far. However, for our particular example, the choice Λ_0^* and its extension to a tree structure provides the best results. In the author's opinion this is due to the fact that one main difficulty of the given optimization problem is, as also indicated in Proposition 3.2, to determine backwards in time the initial data from some given data on the interval. The initial data can only be calculated via a differential equation with negative, large in modulo eigenvalues of $\boldsymbol{A}$, and the choice of Λ_0^* reflects this difficulty.

5 Conclusions

The analytical considerations in this paper suggest the use of a Schur complement as preconditioning. Extreme eigenvalues of symmetric matrices can be eliminated with this technique. The way to choose a good subspace is, although theoretically motivated, still heuristic. Here, the Riesz basis property of wavelets provide the connection with the original problem in function spaces. Knowledge about the origin of the ill-conditioning of this problem may lead to an appropriate coarse choice of Λ_0 based on wavelet compression. Nevertheless, further analysis and investigations are necessary.

Despite that, all numerical experiments, whether listed here or not, indicate successful applications. In our second tested example, a simple choice of Λ_0 reduced the condition number by a factor of $1e + 7$ and in combination with the wavelet-based preconditioning by a factor of $1e + 9$. Obviously, this influences not only the convergence behavior of an iterative method but also the necessary tolerance as stopping criteria for obtaining a sufficient approximation quality. In addition to the extremely small condition numbers the eigenvalue distribution improves too. In most cases large gaps between the smallest eigenvalues arise, which highly favors the efficiency of the pcg-method.

For our application in state estimation, where we are typically faced with very large condition numbers, only the application of the Schur complement technique as a preconditioner enabled us to apply an iterative method in a successful way, providing high estimation quality with low computational cost.

References

[1] O. Abel, A. Helbig, and W. Marquardt. Optimization approaches to control-integrated design of industrial batch reactors. In R. Berber and C. Kravaris, editors, *Nonlinear Model Based Process Control, NATO ASI*, pages 513–551. Kluwer Academic Publishers, Dordrecht, the Netherlands, 1998.

[2] A. Albert and R.W. Sittler. A method for computing least squares estimators that keep up with the data. *SIAM J. Control*, 3:384–417, 1966.

[3] J.S. Albuquerque and L.T. Biegler. Decomposition algorithmus for on–line estimation with nonlinear DAE models. *Comput. Chem. Eng.*, 19:1031–1039, 1995.

[4] J.S. Albuquerque and L.T. Biegler. Data reconcilliation and gross–error detection for dynamic systems. *AIChE J.*, 42(10):2841–2856, 1996.

[5] J.S. Albuquerque, V. Gopal, G.H. Stauss, L.T. Biegler, and B.E. Ydstie. Interior point SQP strategies for structured process optimization problems. *Comput. Chem. Eng.*, 21(Suppl.):853–859, 1997.

[6] F. Allgöwer, T.A. Badgwell, J.S. Qin, J.B. Rawlings, and S.J. Wright. Nonlinear predictive control and moving horizon estimation – an introductory overview. In P.M. Frank, editor, *Advances in Control: Highlights of ECC 1999*, pages 391–449. Springer–Verlag, 1999.

[7] G.A. Almasy. Principles of dynamic balancing. *AIChE J.*, 36:1321–1330, 1990.

[8] H.B. Ameur and B. Kaltenbacher. Regularization of parameter estimation by adaptive discretization using refinement and coarsening indicators. *J. of Inverse and Ill–Posed Problems*, 10(6):561–584, 2002.

[9] O. Axelsson. *Iterative Solution Methods.* Cambridge University Press, Berlin, New York, Cambridge, 1994.

[10] M.J. Bagajewicz and Q. Jiang. Integral approach to plant linear dynamic reconciliation. *AIChE J.*, 43(10):2546–2558, 1997.

[11] B. Bakshi and G. Stephanopoulos. Representation of process trends: III multi–scale extraction of trends from process data. *Comput. Chem. Eng.*, 18(4), 1994.

[12] B. Bakshi and G. Stephanopoulos. Compression of chemical process data by functional approximation and feature extraction. *AIChE J.*, 42(2):477–492, 1996.

[13] J. Bard. *Nonlinear Parameter Estimation.* Academic Press, New York, 1974.

[14] R. Barrett, M. Berry, T.F. Chan, J. Demmel, J. Donato, V. Eijkhout, R. Pozo, C. Romine, and H. Van der Vorst. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods.* SIAM, Philadelphia, PA, 1994.

[15] A. Battermann and M. Heinkenschloss. Preconditioners for Karush-Kuhn-Tucker matrices arising in the optimal control of distributed systems. In W. Desch, F. Kappel, and K. Kunisch, editors, *Control and Estimation of Distributed Parameter Systems*, ISNM 126, pages 15–32. Birkhäuser, Basel, 1998.

[16] J.T. Betts. Survey of numerical methods for trajectory optimization. *J. Guidance, Control, and Dynamics*, 21(2):193–207, 1998.

[17] J.T. Betts. *Practical Methods for Optimal Control Using Nonlinear Programming.* SIAM, Philadelphia, 2001.

[18] L.T. Biegler. Efficient solution of dynamic optimization and NMPC problems. In F. Allgöwer and A. Zheng, editors, *Nonlinear Model Predictive Control*, pages 219–243. Birkhäuser Verlag, Basel, Switzerland, 2000.

[19] T. Binder, L. Blank, H.G. Bock, R. Bulirsch, W. Dahmen, M. Diehl, T. Kronseder, W. Marquardt, J.P. Schlöder, and O. van Stryk. Introduction to model based optimization of chemical processes on moving horizons. In M. Grötschel, S.O. Krumke, and J. Rambau, editors, *Online Optimization of Large Scale Systems*, pages 295–339. Springer–Verlag, Berlin Heidelberg, 2001.

[20] T. Binder, L. Blank, W. Dahmen, and W. Marquardt. Towards multiscale dynamic and data reconciliation. In R. Berber and C. Kravaris, editors, *Nonlinear Model Based Process Control, NATO ASI*, pages 623–665. Kluwer Academic Publishers, Dordrecht, the Netherlands, 1998.

[21] T. Binder, L. Blank, W. Dahmen, and W. Marquardt. Adaptive multiscale approach to real–time dynamic optimization. LPT Report, LPT–1999–14, RWTH Aachen, 1999.

[22] T. Binder, L. Blank, W. Dahmen, and W. Marquardt. An adaptive multiscale method for real–time moving horizon optimization. In *Proc. American Control Conference 2000*, pages 4234–4238. Omnipress, Madison, Wisconsin, 2000.

[23] T. Binder, L. Blank, W. Dahmen, and W. Marquardt. Grid refinement in multiscale dynamic optimization. In S. Pierucci, editor, *Proceedings of the European Symposium on Computer Aided Process Engineering, (ESCAPE-10)*, pages 31–36. Elsevier, New York, 2000.

[24] T. Binder, L. Blank, W. Dahmen, and W. Marquardt. Iterative algorithms for multiscale dynamic data reconciliation. IGPM–Report 186, RWTH Aachen, 2000.

[25] T. Binder, L. Blank, W. Dahmen, and W. Marquardt. Regularization of dynamic data reconciliation problems by projection. In L.T. Biegler, A. Brambilla, and C. Scali, editors, *Proc. IFAC International Symposium on Advanced Control of Chemical Processes, ADCHEM 2000*, volume 2, pages 689–694. IFAC – International Federation of Automatic Control, 2000.

[26] T. Binder, L. Blank, W. Dahmen, and W. Marquardt. Iterative algorithms for multiscale state estimation, part 1: Concept. *J. Optim. Theory Appl.*, 111(3):501–527, 2001.

[27] T. Binder, L. Blank, W. Dahmen, and W. Marquardt. Iterative algorithms for multiscale state estimation, part 2: Numerical investigation. *J. Optim. Theory Appl.*, 111(3):529–551, 2001.

[28] T. Binder, L. Blank, W. Dahmen, and W. Marquardt. Multiscale concepts for moving horizon optimization. In M. Grötschel, S.O. Krumke, and J. Rambau, editors, *Online Optimization of Large Scale Systems*, pages 341–361. Springer–Verlag, Berlin, Heidelberg, 2001.

[29] T. Binder, L. Blank, W. Dahmen, and W. Marquardt. On the regularization of dynamic data reconciliation problems. *Journal of Process Control*, 12:557–567, 2002.

[30] T. Binder, A. Cruse, C. Villar, and W. Marquardt. Optimization using a wavelet based adaptive control vector parametrization strategy. *Comput. Chem. Eng.*, 24:1201–1207, 2000.

[31] J. Birk and M. Zeitz. Extended Luenberger observer for non–linear multivariable systems. *Int. J. Control*, 47(6):1823–1836, 1988.

[32] L. Blank. Preconditioning via a Schur complement method: an application in state estimation. to be published in SIAM J. Sci. Comput., 2003.

[33] L. Blank. State estimation without regularizing the initial data. Preprint, 2003.

[34] H.G. Bock, M.M. Diehl, D.B. Leineweber, and J.P. Schlöder. A direct multiple shooting method for real–time optimization of nonlinear DAE processes. In F. Allgöwer and A. Zheng, editors, *Nonlinear Model Predictive Control*, pages 245–267. Birkhäuser Verlag, Basel, Switzerland, 2000.

[35] D. Braess. *Finite Elemente*. Springer–Verlag, Berlin, 1992.

[36] J.H. Bramble and J.E. Pasciak. A preconditioning technique for indefinite systems resulting from mixed approximations for elliptic problems. *Math. Comp.*, 50:1–17, 1988.

[37] K.E. Brenan, S.L. Campbell, and L.R. Petzold. *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations.* SIAM, Philadelphia, 1996.

[38] F. Brezzi and M. Fortin. *Mixed and Hybrid Finite Element Methods.* Springer–Verlag, Berlin, 1991.

[39] A.G. Bruce, D.L. Donoho, H.Y. Gao, and R.D. Martin. Smoothing and robust wavelet analysis. In *Proceedings on Computational Statistics. 11th Symposium, Berlin*, volume 4, pages 531–547. COMPSTAT, 1994.

[40] A. Cervantes and L.T. Biegler. Large–scale DAE optimization using a simultaneous NLP formulation. *AIChE J.*, 44(5):1038–1050, 1998.

[41] A. Chambolle, R. DeVore, N.Y. Lee, and J. Lucier. Nonlinear wavelet image processing: Varational problems, compression, and noise removal through wavelet shrinkage. *IEEE Image Processing*, 7:319–335, 1998.

[42] Z. Cheng and W.D. Pilkey. Wavelet–based limiting performance analysis of mechanical systems subject to transient disturbances. *Finite Elements in Analysis and Design*, 33:233–245, 1999.

[43] C.K. Chui. *An Introduction to Wavelets.* Academic Press, Boston, Massachusetts, 1992.

[44] C.K. Chui. *Wavelets: A Mathematical Tool for Signal Analysis.* SIAM, Philadelphia, PA, 1997.

[45] A. Cohen, W. Dahmen, and R.A. DeVore. Adaptive wavelet methods II. Beyond the elliptic case. *Found. Comput. Math.*, 2(3):203–245, 2002.

[46] A. Cohen, I. Daubechies, and J.C. Feauveau. Biorthogonal bases of compactly supported wavelets. *Comm. Pure and Appl. Math.*, 45:485–560, 1992.

[47] A. Cohen and R. Masson. Adaptive wavelet methods for second order elliptic problems, preconditioning and adaptivity. *SIAM J. Sci. Comp.*, 21:1006–1026, 1999.

[48] A. Cohen and R.D. Ryan. *Wavelets and Multiscale Signal Processing.* Chapman and Hall, London, 1995.

[49] A.R. Conn, N. Gould, and Ph. Toint. *Trust–Region Methods.* SIAM, Philadelphia, 2000.

[50] M.J. Corless and A.E. Frazho. *Linear Systems and Control, An Operator Perspective.* Marcel Dekker, Inc., New York, Basel, 2003.

[51] H. Cox. On the estimation of state variables and parameters for noisy dynamic systems. *IEEE Trans. Auto. Cont.*, pages 5–12, 1964.

[52] C.M. Crowe. Data reconciliation – progress and challage. *JPC*, 6:89–98, 1996.

[53] R.F. Curtain and H. Zwart. *An introduction to infinite–dimensional linear systems theory.* Springer–Verlag, New York, 1995.

[54] J.E. Cuthrell and L.T. Biegler. On the optimization of differential–algebraic process systems. *AIChE J.*, 33(8):1257–1270, 1987.

[55] S.A. Dadebo and K.B. McAuley. Dynamic optimization of constrained chemical engineering problems using dynamic programming. *Comput. Chem. Eng.*, 19(5):513–525, 1995.

[56] W. Dahmen. Wavelet and multiscale methods for operator equations. *Acta Numerica*, 7:55–228, 1997.

[57] W. Dahmen and A. Kunoth. Multilevel preconditioning. *Numer. Math.*, 63(2):315–344, 1992.

[58] W. Dahmen, A. Kunoth, and K. Urban. Wavelets in numerical analysis and their quantitative properties. In A. Le Méhauté, C. Rabut, and L.L. Schumaker, editors, *Surface Fitting and Multiresolution Methods*, pages 93–130. Vanderbilt University Press, 1997.

[59] W. Dahmen, A. Kunoth, and K. Urban. Biorthogonal spline–wavelets on the interval – stability and moment conditions. *Applied and Computational Harmonic Analysis*, 6:132–196, 1999.

[60] W. Dahmen and C.A. Micchelli. Using the refinement equation for evaluating integrals of wavelets. *SIAM J. Numer. Anal.*, 30(2):507–537, 1993.

[61] W. Dahmen, S. Prössdorf, and R. Schneider. Multiscale methods for pseudodifferential equations on smooth manifolds. In C.K. Chui, L. Montefusco, and L. Puccio, editors, *Proceedings of the International Conference on Wavelets: Theory, Algorithms, and Applications*, pages 385–424. Academic Press, San Diego, 1994.

[62] W. Dahmen, R. Schneider, and Y. Xu. Nonlinear functionals of wavelet expansions – adaptive reconstruction and fast evaluation. *Numer. Math.*, 86(1):49–101, 2000.

[63] M. Darouach and M. Zasadzinski. Data reconciliation in generalized linear dynamic systems. *AIChE J.*, 37:193–201, 1991.

[64] I. Daubechies. *Ten Lectures on Wavelets.* SIAM, Philadelphia, Pennsylvania, 1992.

[65] C. de Boor. *A Practical Guide to Splines.* Springer–Verlag, New York, 1978.

[66] S. de Oliveira and M. Morari. Contractive model predictive control with local linearization for nonlinear systems. In R. Berber and C. Kravaris, editors, *Nonlinear Model Based Process Control, NATO ASI*, pages 403–432. Kluwer Academic Publishers, 1998.

[67] P. de Vallière and D. Bonvin. Application of estimation techniques to batch reactors – III: Modelling refinements which improve the quality of state and parameter estimation. *Comput. Chem. Eng.*, 14(7):799–808, 1990.

[68] R.A. DeVore, B. Jawerth, and B. Lucier. Image compression through wavelet transform coding. *IEEE Trans. Information Theory*, 38(2):719–746, 1992.

[69] R.A. DeVore, B. Jawerth, and V.A. Popov. Compression of wavelet decompositions. *Amer. J. Math.*, 114:737–785, 1992.

[70] V. Dicken and P. Maass. Wavelet–Galerkin methods for ill–posed problems. *J. Inv. Ill–Posed Problems*, 4(3):203–222, 1996.

[71] D. Donoho. Nonlinear solution of linear inverse problems by Wavelet–Vaguelette decomposition. *Appl. Comp. Harmon. Anal.*, 2:101–126, 1995.

[72] D. Donoho and I. Johnstone. Ideal spatial adaption by wavelet shrinkage. *Biometrika*, 81:425–455, 1994.

[73] D. Donoho, I. Johnstone, G. Kerkyacharian, and D. Picard. Wavelet shrinkage: Asymptopia? *J. Roy. Statist. Assoc.*, 90:301–369, 1995.

[74] D.R.Kuehn and H.Davidson. Computer control. *Chemical Engineering Progress*, 57(6):44–47, 1961.

[75] H.C. Elman, D.J. Silvester, and A.J. Wathen. Performance and analysis of saddle point preconditioners for the discrete steady–state Navier–Stokes equations. *Numer. Math.*, 90(4):665–688, 2002.

[76] H.W. Engl. Regularization methods for the stable solution of inverse problems. *Surv. Math. Ind.*, 3:71–143, 1993.

[77] H.W. Engl, M. Hanke, and A. Neubauer. *Regularization of Inverse Problems.* Kluwer, Dordrecht, The Netherlands, 1996.

[78] W.F. Feehery and P.I. Barton. Dynamic simulation and optimization with inequality path constraints. *Comput. Chem. Eng.*, 20:707–712, 1996.

[79] R. Fletcher. *Practical Methods of Optimization.* John Wiley & Sons, Chichester, New York, Brisbane,Toronto, Singapore, 2001.

[80] J. Frank and C. Vuik. On the construction of deflation–based preconditioners. *SIAM J. Sci. Comput.*, 23(2):442–462, 2001.

[81] W. Freeden and F. Schneider. Regularization wavelets and multiresolution. *J. Inv. Ill–Posed Problems*, 14:225–243, 1998.

[82] R.W. Freund, G.H. Golub, and N.M. Nachtigal. Iterative solution of linear systems. *Acta Numerica*, 2:57–100, 1992.

[83] J. Fröhlich and K. Schneider. An adaptive wavelet–vaguelette algorithm for the solution of nonlinear PDEs. Preprint SC. 95-28, ZIB, Berlin, 1997.

[84] G.F. Froment and K.B. Bischoff. *Chemical Reactor Analysis and Design*. John Wiley & Sons, 1990.

[85] A. Gelb. *Applied Optimal Estimation*. MIT Press, Cambridge, 1974.

[86] J. Gertler and G.A. Almasy. Balance calculation through dynamic system modeling. *Automatica*, 9:79–85, 1973.

[87] P.E. Gill, M. Murray, and M.A. Saunders. User's guide for SNOPT 5.3: a Fortran package for large–scale nonlinear programming. Report na 97-5, Department of Mathematics, University of California, San Diego, 1997.

[88] R. Glowinski. *Numerical Methods for Nonlinear Variational Problems*. Springer–Verlag, New York, 1984.

[89] G.H. Golub and C.F. Van Loan. *Matrix Computations*. The John Hopkins University Press, Baltimore, MD, 3rd edition, 1996.

[90] A. Greenbaum. *Iterative Methods for Solving Linear Systems*. SIAM, Philadelphia, 1997.

[91] M. Grötschel, S.O. Krumke, and J. Rambau. *Online Optimization of Large Scale Systems*. Springer–Verlag, Berlin Heidelberg, 2001.

[92] M. Hanke. *Conjugate gradient type methods for ill–posed problems*. Longman Scientific and Technical, Harlow, Essex, 1995.

[93] C. Hansen. *Rank–Deficient and Discrete Ill–posed Problems: Numerical Aspects of Linear Inversion*. SIAM, Philadelphia, 1998.

[94] H. Harbrecht, S. Pereverzev, and R. Schneider. An adaptive regularization by projection for noisy pseudodifferential equations of negative order. Preprint SFB 393 01-31, TU Chemnitz, 2001.

[95] R.F. Hartl, S.P. Sethi, and R.G. Vickson. A survey of the maximum principles for optimal control problems with state constraints. *SIAM Review*, 37(2):181–218, 1995.

[96] M. Heinkenschloss and F. Tröltzsch. Analysis of the Lagrange–SQP–Newton method for the the control of a phase field equation. *Control and Cybernetics*, 28(2):178–211, 1999.

[97] M.A. Henson and D.E. Seborg. *Nonlinear Process Control.* Prentice Hall, Upper Saddle River, 1997.

[98] M.R. Hestenes. *Calculus of Variations and Optimal Control Theory.* John Wiley & Sons, Inc., New York, 1966.

[99] D.M. Himmelblau and T.W. Karjala. Rectification of data in a dynamic process using artificial neural networks. *Comput. Chem. Eng.*, 20:805–812, 1996.

[100] M. Hinze and K. Kunisch. Second order methods for optimal control of time–dependent fluid flow. *Siam J. Control and Optim.*, 40:925–946, 2001.

[101] M. Hinze and K. Kunisch. Second order methods for boundary control of the instationary Navier–Stokes system. *ZAMM*, to appear.

[102] R.M. Hirschhorn. Invertability of multivariable nonlinear control systems. *IEEE Trans. Auto. Cont.*, 24(6):855–865, 1979.

[103] K.-H. Hoffmann, R.H.W. Hoppe, and V. Schulz, editors. *Fast Solution of Discretized Optimization Problems.* ISNM 138. Birkäuser Verlag, 2001.

[104] C.H. Hsiao and W.J. Wang. Optimal control of time–varying systems via Haar wavelets. *J. Optim. Theory Appl.*, 103:641–655, 1999.

[105] K. Ito and K. Kunisch. Augmented Lagrangian–SQP methods for nonlinear optimal control problems of tracking type. *SIAM J. Control and Optimization*, 34:874–891, 1996.

[106] S.-S. Jang, B. Joseph, and H. Mukal. Comparison of two approaches to on-line parameter and state estimation of nonlinear systems. *Ind. Eng. Chem. Proc. Des. Dev.*, 25:809–814, 1986.

[107] A.H. Jazwinski. *Stochastic Processes and Filtering Theory.* Academic Press, New York, 1970.

[108] T. Johansen. On Tikhonov regularization, bias and variance in nonlinear system identification. *Automatica*, 33(3):441–446, 1997.

[109] J.V. Kadam. Index analysis and adaptive refinement in multiscale dynamic optimization. Master thesis, Lehrstuhl für Prozesstechnik, RWTH Aachen, 2000.

[110] T. Kailath. *Linear Systems.* Prentice Hall, Englewood Cliffs, New Jersey, 1980.

[111] R.E. Kalman. A new approach to linear filtering and prediction problems. *Trans. ASME J. Basic Eng.*, 82:34–45, 1960.

[112] R.E. Kalman and R.S. Bucy. New results in linear filtering and prediction theory. *Trans. ASME J. Basic Eng.*, 83:95–108, 1961.

[113] T.W. Karjala and D.M. Himmelblau. Dynamic data rectification by recurrent neural networks vs. traditional methods. *AIChE J.*, 40(11):1865–1875, 1994.

[114] J. Kelly. A regularization approach to the reconciliation of constrained data sets. *Comput. Chem. Eng.*, 12(12):1771–1778, 1998.

[115] I.-W. Kim, M.J. Liebmann, and T.F.Edgar. A sequential error-in variables method for non-linear dynamic systems. *Comput. Chem. Eng.*, 15:663–670, 1991.

[116] I.-W. Kim, S.W. Park, and T.F.Edgar. Data reconciliation for input-output models in nonlinear dynamic systems. *Korean J. Chem. Engng*, 13:211–215, 1996.

[117] A. Kirsch. *An Introduction to the Mathematical Theory of Inverse Problems.* Springer–Verlag, 1996.

[118] H.W. Knobloch and H. Kwakernaak. *Lineare Kontrolltheorie.* Springer–Verlag, Berlin, Heidelberg, 1985.

[119] J.L.A. van Koolen. Plant operation in the future. *Comput. Chem. Eng.*, 18:S477–S481, 1994.

[120] A. Kunoth. Multilevel preconditioning – appending boundary conditions by Lagrange multipliers. *Advances in Computational Mathematics*, 4:145–170, 1995.

[121] H. Kwakernaak and R. Sivan. *Linear Optimal Control Systems.* Wiley-Interscience, John Wiley & Sons, Inc., New York, 1972.

[122] J. Lee, Y. Chikkula, Z. Yu, and J. Kantor. Improving computational efficiency of model predictive control algorithm using wavelet transformation. *Int. J. Control*, 61(4):859–883, 1995.

[123] M.J. Liebmann, T.F. Edgar, and L.S. Lasdon. Efficient data reconciliation and estimation for dynamic processes using nonlinear programming techniques. *Comput. Chem. Eng.*, 16:963–986, 1992.

[124] L.C. Lin and C.-C. Kuo. Signal extrapolation in noisy data with wavelet representation. In *Proceedings of the SPIE*, volume 2028, pages 156–167. The International Society for Optical Engineering, 1993.

[125] J.S. Logsdon and L.T. Biegler. Decomposition strategies for large–scale dynamic optimization problems. *Chem. Eng. Sci.*, 47(4):851–864, 1992.

[126] D.G. Luenberger. Observing the state of a linear system. *IEEE Trans. Mil. Electron.*, 8:74–80, 1964.

[127] D.G. Luenberger. Observers for multivariable systems. *IEEE Trans. Autom. Control.*, 11(2):190–197, 1966.

[128] P. Maaß and A. Rieder. Wavelet–accelerated Tikhonov–Philipps regularization with applications. In H.W. Engl, A.K. Louis, and W. Rundell, editors, *Inverse Problems in Medical Imaging and Nondestructive Testing*, pages 134–158. Springer–Verlag, Wien, 1997.

[129] S. Mallat. *A Wavelet Tour of Signal Processing.* Academic Press, 1999.

[130] L. Mansfield. Damped Jacobi preconditioning and coarse grid deflation for conjugate gradient iteration on parallel computers. *SIAM J. Sci. Statist. Comput.*, 12:1314–1323, 1991.

[131] W. Marquardt. Numerical methods for the simulation of differential–algebraic process models. In R. Berber, editor, *Methods of Model-Based Control*, NATO-ASI Series, pages 42–79. Kluwer Press, 1995.

[132] K.F. McBrayer, T.F. Edgar, and L.S. Lasdon. Bias detection and estimation in dynamic data reconciliation. *Journal of Process Control*, 5:285–289, 1995.

[133] A. Mertins. *Signal Analysis: Wavelets, Filter Banks, Time-Frequency Transforms and Applications.* John Wiley & Sons, Chichester, 1999.

[134] G. Meurant. *Computer Solution of Large Linear Systems.* North–Holland, Amsterdam, the Netherlands, 1999.

[135] A. Mhamdi, A. Helbig, O. Abel, and W. Marquardt. Newton–type receding horizon control and state estimation – a case study. In J.J. Gertler, J.B. Cruz, and M. Peshkin, editors, *Preprints of the 13th World Congress of IFAC*, volume M, pages 121–126, San Francisco, 1996.

[136] A. Mhamdi and W. Marquardt. An inversion approach to the estimation of reaction rates in chemical reactors. In *Proceedings of the 5th European Control Conference, (ECC'99).* Springer–Verlag, Karlsruhe, Germany, 1999.

[137] H. Michalska and D.Q. Mayne. Moving horizon observers. In *IFAC Symposium NOLCOS'92*, Bordeaux, France, 1992.

[138] H. Michalska and D.Q. Mayne. Moving horizon observers and observer–based control. *IEEE Trans. Aut. Contr.*, 40(6):995–1006, 1995.

[139] M. Morari and J.H. Lee. Model predictive control: Past, present and future. *Comput. Chem. Engng.*, 23:667–682, 1999.

[140] R.B. Morgan. A restarted GMRES method augmented with eigenvectors. *SIAM J. Matrix Anal. Appl.*, 16(4):1154–1171, 1995.

[141] V. A. Morozov. *Methods for solving incorrectly posed problems.* Springer–Verlag, New York, 1984.

[142] A.I. Mouaddib and S. Zilberstein. Knowledgebased anytime computation. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, volume 1, pages 775–781, 1995.

[143] K.R. Muske and T.F. Edgar. Nonlinear state estimation. In M.A. Henson and D.E. Seborg, editors, *Nonlinear Process Control*, pages 311–711. Prentice Hall, 1997.

[144] K.R. Muske and J.B. Rawlings. Nonlinear moving horizon state estimation. In R. Berber, editor, *Methods of Model–Based Control*, pages 349–365. NATO–ASI Series, Kluwer Press, Dordrecht, The Netherlands, 1995.

[145] S. Narasimhan and C. Jordache. *Data Reconciliation and Gross Error Detection.* Gulf Publishing Company, Houston, Texas, 2000.

[146] A. Neumaier. Solving ill–conditioned and singular linear systems: A tutorial on regularization. *SIAM Rev.*, 40:636–666, 1998.

[147] R.A. Nicolaides. Deflation of conjugate gradients with applications to boundary value problems. *SIAM J. Numer. Anal.*, 24:355–365, 1987.

[148] J. Nocedal and S.J. Wright. *Numerical Optimization.* Springer–Verlag, New–York, 1999.

[149] C.C. Paige and M.A. Saunders. Solution of sparse indefinite systems of linear equations. *SIAM J. Numer. Anal.*, 12:671–629, 1975.

[150] A. Petrosian and F.G. Meyer. *Wavelets in Signal and Image Analysis.* (editors), Kluwer Academic Publishers, 2001.

[151] R. Plato and G. Vainikko. On the regularization of projection methods for solving ill–posed problems. *Numer. Math.*, 57:63–79, 1990.

[152] L.S. Pontryagin, V.G. Boltyanski, R.V. Gamkrelidze, and E.F. Miscenko. *The Mathematical Theory of Optimal Processes.* Wiley, 1962.

[153] R. Pytlak. *Numerical Methods for Optimal Control Problems with State Constraints.* Springer–Verlag, 1999.

[154] Y. Ramamurthi, P.B. Sistu, and B.W. Bequette. Control–relevant dynamic data reconciliation and parameter estimation. *Comput. Chem. Eng.*, 17(1):41–59, 1993.

[155] C.V. Rao and J.B. Rawlings. Nonlinear moving horizon estimation. In F. Allgöwer and A. Zheng, editors, *Nonlinear Model Predictive Control, Vol. 26 of Progress in Systems and Control Theory*, pages 45–69. Birkhäuser Verlag, Basel, 2000.

[156] C.V. Rao and J.B. Rawlings. Constrained process monitoring: Moving–horizon approach. *AIChE J.*, 48(1):97–109, 2002.

[157] D. Robertson, J.H. Lee, and J.B. Rawlings. A moving horizon–based approach for least–squares estimation. *AIChE J.*, 42(8):2209–2223, 1996.

[158] J. A. Romagnoli and M. C. Sanchez. *Data Processing and Reconciliation for Chemical Process Operation.* Academic Press, San Diego, 1999.

[159] Y. Saad. Analysis of augmented Krylov subspace methods. *SIAM J. Matrix Anal. Appl.*, 18(2):435–449, 1997.

[160] T. Samad and D. Cofer. Autonomy in automation: trends, technologies, tools. In R. Gani and S.B. Jorgensen, editors, *European Symposium on Computer Aided Process Engineering - 11*, pages 1–13. Elsevier, 2001.

[161] K. Schittkowski. Solving constraint nonlinear least squares problems by a general purpose SQP–method. *International Series of Numerical Mathematics*, 84:295–309, 1988.

[162] C. Schmid and L.T. Biegler. Quadratic programming methods for reduced hessian SQP. *Comput. Chem. Eng.*, 18(9):817–832, 1994.

[163] C. Schmid, J. Nocedal, and L.T. Biegler. A reduced hessian method for large–scale constrained optimization. *SIAM J. Optimization*, 5(2):314, 1995.

[164] J.G. Schmidt and G. Starke. Coarse space orthogonalization for indefinite linear systems of equations arising in geometrically nonlinear elasticity. *Appl. Numer. Math.*, 30:267–280, 1999.

[165] H. Schuler and C. Schmidt. Calorimetric state estimators for chemical reactor diagnosis and control: review of methods and applications. *Chem. Eng. Sci.*, 47:899–915, 1992.

[166] V. Schulz. SQP–based direct discretization methods for practical optimal control problems. *Journal of Computational and Applied Mathematics*, Special issue (guest editor), 120(1-2), 2000.

[167] L.M. Silverman. Inversion of multivariable linear systems. *IEEE Transactions on Automatic Control*, AC–14(3):270–276, 1969.

[168] P.B. Sistu, R.S. Gopinath, and J.B. Rawlings. Computional issues in nonlinear predictive control. *Comput. Chem. Eng.*, 17:361, 1993.

[169] G. Stephanopoulos, M. Dyer, and O. Karsligil. Multi–scale modeling, estimation and control of processing systems. *Comput. Chem. Eng.*, 21(Suppl.):797–803, 1997.

[170] G. Stephanopoulos, M. Dyer, and O. Karsligil. Multiscale aspects in model predictive control. In R. Berber and C. Kravaris, editors, *Nonlinear Model Based Process Control, NATO ASI*, pages 667–734. Kluwer Academic Publishers, Dordrecht, the Netherlands, 1998.

[171] P. Tanartkit and L.T. Biegler. Stable decomposition for dynamic optimization. *Ind. Eng. Chem. Res.*, 34(4):1253–1266, 1995.

[172] D. Tieu, W.R. Cluett, and A. Penlidis. A comparison of collocation methods for solving dynamic optimization problems. *Comput. Chem. Eng.*, 19(4):375–381, 1995.

[173] A.N. Tikhonov and V. Arsenin. *Solutions of Ill–posed Problems.* Wiley, New York, New York, 1977.

[174] A.N. Tikhonov, A.V. Goncharsky, V.V. Stepanov, and A.G. Yagola. *Numerical Methods for the Solution of Ill–posed Problems.* Kluwer Academic Publishers, New York, 1996.

[175] F. Tröltzsch. On the Lagrange–Newton–SQP method for the optimal control of semilinear parabolic equations. *SIAM J. Control and Optimization*, 38:294–312, 1999.

[176] M.L. Tyler and M. Morari. Stability of constrained moving horizon estimation schemes. In R. Berber, editor, *Nonlinear Model Based Process Control*, NATO-ASI Series. Kluwer Press, 1998.

[177] J. Unger, A. Kröner, and W. Marquardt. Structural analysis of differential–algebraic equation systems – theory and applications. *Comput. Chem. Eng.*, 19(8):867–882, 1995.

[178] A. Uppal, W. H. Ray, and A. B. Poore. On the dynamic behaviour of continuous stirred tank reactors. *Chem. Eng. Sci.*, 29:967–985, 1974.

[179] A. van der Sluis and H.A. van der Vorst. The rate of convergence of conjugate gradients. *Numer. Math.*, 48:543–560, 1986.

[180] V.S. Vassiliadis, R.W.H. Sargent, and C.C. Pantelides. Solution of a class of multistage dynamic optimization problems. 1. Problems without path constraints. *Ind. Eng. Chem. Res.*, 33(9):2111–2122, 1994.

[181] V.S. Vassiliadis, R.W.H. Sargent, and C.C. Pantelides. Solution of a class of multistage dynamic optimization problems. 2. Problems with path constraints. *Ind. Eng. Chem. Res.*, 33(9):2123–2133, 1994.

[182] O. von Stryk. *Numerische Lösung optimaler Steuerungsprobleme: Diskretisierung, Parameteroptimierung und Berechnung der adjungierten Variablen.* Reihe 8: Meß–, Steuerungs– und Regelungstechnik Nr. 441. VDI–Verlag, Düsseldorf, 1995.

[183] C. Vuik, A. Segal, and J.A. Meijerink. An efficient preconditioned CG method for the solution of a class of layered problems with extreme contrasts in the coefficients. *J. Comput. Phys.*, 152:1–19, 1999.

[184] B. Walcott, M. Corless, and S. Zak. Comparative study of non–linear state–observation techniques. *Int. J. Control*, 45(6):2109–2132, 1987.

[185] W. Waldraff, D. Dochain, S. Bourrel, and A. Magnus. On the use of observability measures for sensor location in tubular reactors. *Journal of Process Control*, 8:497–505, 1998.

[186] C. Walter. Implementation and evaluation of interior–point–methods for the solution of wavelet discretized quadratical optimal control problems. Senior project, LPT, RWTH Aachen, 1999.

[187] W. Walter. *Gewöhnliche Differentialgleichungen.* Springer–Verlag, Berlin, 1976.

[188] G.H. Weiss, J.A. Romagnoli, and K.A. Islam. Data reconciliation – an industrial case study. *Comput. Chem. Eng.*, 20:1441–1449, 1996.

[189] J.H. Wilkinson. *The Algebraic Eigenvalue Problem.* Oxford University Press, London, 1965.

[190] S. Wright. *Primal–Dual Interior–Point Methods.* SIAM, 1997.

[191] M. Zeitz. Transfer characteristics of extremely fast state feedback and observer systems. *Int. J. Systems Science*, 14:167–177, 1983.

[192] D. Zhou, W. Cai, and W. Zhang. An adaptive wavelet method for nonlinear circuit simulation. *IEEE Transactions on Circuits and Systems–1: Fundamental Theory and Applications*, 46(8):931–938, 1999.