# Iterative Algorithms for Multiscale State Estimation, Part 1: Concept*

T. Binder[1], L. Blank[2], W. Dahmen[2], W. Marquardt[1]

[1] Lehrstuhl für Prozesstechnik,
[2] Institut für Geometrie und Praktische Mathematik,
RWTH Aachen, Germany.

April, 2001

to appear in

Journal of Optimization Theory and Applications

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

**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 problem*, where the difference between modeled outputs $\boldsymbol{y}$ and measured data $\boldsymbol{z}$ on a receding horizon is to be minimized (Refs. 1–3). 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 (Refs. 4–5) and the generalized inverse is discontinuous (Ref. 6). In the context of dynamic data reconciliation (Ref. 4) 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 (Ref. 7). For a more detailed discussion of the form of the objective function we refer to (Ref. 4).

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 (Refs. 8–9). 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 (Ref. 10) 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 $n_w \leq n_x$, $n_v \leq n_y$, the resulting linear optimization problem has the form

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

subject to the constraints

$$\begin{aligned} \dot{\boldsymbol{x}} - \boldsymbol{A}\boldsymbol{x} - \boldsymbol{W}\boldsymbol{w} &= \boldsymbol{B}\boldsymbol{u}, \qquad t \in [0,1], & (2)\\ \boldsymbol{y} - \boldsymbol{C}\boldsymbol{x} - \boldsymbol{V}\boldsymbol{v} &= \boldsymbol{0}. & (3) \end{aligned}$$

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 (Ref. 11). The right choice of the regularization terms $\boldsymbol{R_v}$ and $\boldsymbol{R_w}$ is still open at least for nonlinear problems (see Ref. 3 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

$$\begin{aligned} \langle \boldsymbol{\zeta}_1, \dot{\boldsymbol{x}} - \boldsymbol{A}\boldsymbol{x} - \boldsymbol{W}\boldsymbol{w}\rangle &= \langle \boldsymbol{\zeta}_1, \boldsymbol{B}\boldsymbol{u}\rangle \qquad \text{for all } \boldsymbol{\zeta}_1 \in (L_2)^{n_x} & (4)\\ \langle \boldsymbol{\zeta}_2, \boldsymbol{y} - \boldsymbol{C}\boldsymbol{x} - \boldsymbol{V}\boldsymbol{v}\rangle &= \boldsymbol{0} \qquad\qquad\quad \text{for all } \boldsymbol{\zeta}_2 \in (L_2)^{n_y}. & (5) \end{aligned}$$

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 (1) and writing $b(\boldsymbol{\zeta}, \boldsymbol{\Upsilon})$ for the combined bilinear forms resulting from the left hand side of (4), (5), 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 (1)–(3) 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

$$\begin{aligned} a(\boldsymbol{\Upsilon}, \boldsymbol{\Upsilon}') + b(\boldsymbol{\lambda}, \boldsymbol{\Upsilon}') &= \langle 2\boldsymbol{Q}\boldsymbol{z}, \boldsymbol{\Upsilon}' \rangle \quad \text{for all } \boldsymbol{\Upsilon}' \in \Upsilon, && (6) \\ b(\boldsymbol{\zeta}, \boldsymbol{\Upsilon}) &= \langle \boldsymbol{\zeta}_1, \boldsymbol{B}\boldsymbol{u} \rangle \quad \text{for all } \boldsymbol{\zeta} \in \mathcal{M}. && (7) \end{aligned}$$

In view of the Riesz representation theorem the weak formulation (6)–(7) is equivalent to the operator equation

$$\boldsymbol{\mathcal{L}} \begin{pmatrix} \boldsymbol{\Upsilon} \\ \boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} \boldsymbol{h}(\boldsymbol{z}, \boldsymbol{u}) \\ \mathrm{g}(\boldsymbol{z}, \boldsymbol{u}) \end{pmatrix} \tag{8}$$

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 (1)–(3) is well–posed means that the mapping $\mathcal{L}$ 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

$$c\left(\|\boldsymbol{\Upsilon}\|_\Upsilon^2 + \|\boldsymbol{\lambda}\|_\mathcal{M}^2\right) \leq \|\mathcal{L}\begin{pmatrix} \boldsymbol{\Upsilon} \\ \boldsymbol{\lambda} \end{pmatrix}\|^2_{((H^1)^{n_x})' \times (L_2^{n_x+2n_y+n_v+n_w})} \leq \tilde{c}\left(\|\boldsymbol{\Upsilon}\|_\Upsilon^2 + \|\boldsymbol{\lambda}\|_\mathcal{M}^2\right). \tag{9}$$

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. $\boldsymbol{R_v} = \boldsymbol{0}, \boldsymbol{R_w} = \boldsymbol{0}$) may be ill–posed and the way of regularization affects the constants in (9). 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 Ref. 12). 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 Refs. 13–14 and references therein). In the context of simulation and dynamic optimization of ordinary differential equations several related wavelet approaches appeared recently in the literature (Refs. 15–17). 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 (6)–(7) of the optimization problem (1)–(3) with a Galerkin method we have to choose appropriate finite dimensional subspaces $\Upsilon_\Lambda$ and $\mathcal{M}_\Lambda$ of $\Upsilon$ 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 (Refs. 14,18,19).

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 (4) 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 $[0,1]$ (Ref. 20). 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 $\Upsilon$ 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 (2) are tested by Haar wavelets $\psi^H_{j,k}$ and the output equations (3) 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 (6)–(7). 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 (6)–(7) to such choices of subspaces $\Upsilon_\Lambda$ of $\Upsilon$ and $\mathcal{M}_\Lambda$ of $\mathcal{M}$, exhibit the sparseness structure illustrated by Figure 1. Of course, $\Lambda$ is comprised here of finite index sets $\Lambda_{x_i}$, $\Lambda_{y_i}$, $\Lambda_{v_i}$ and $\Lambda_{w_i}$. 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 $\Lambda_x$, all outputs the index set $\Lambda_y$ and likewise for the index sets of the remaining groups of components. Moreover, to ensure stability of the discretizations the spaces $\Upsilon_\Lambda$, $\mathcal{M}_\Lambda$ have to satisfy the LBB-condition (Ref. 21). This means that $\mathcal{M}_\Lambda$ may not be too rich relative to $\Upsilon_\Lambda$. Specifically, this requires that the index sets $\Lambda'_x$ for the first group of Lagrange multipliers $\boldsymbol{\mu}$ acting on (2) should be contained in $\Lambda_x$. 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 (Ref. 22).

Let us collect now in $\boldsymbol{d_x}$ all wavelet coefficients of the states $\boldsymbol{x}_\Lambda$ ($\boldsymbol{d_x} = \left(\boldsymbol{d}^\Lambda_{\boldsymbol{x}_i}\right)_{i=1,\ldots n_x}$) and likewise for $\boldsymbol{d_y}, \boldsymbol{d_v}, \boldsymbol{d_w}$ where we have suppressed for convenience the index $\Lambda$. Inserting now the ansatz $\boldsymbol{\Upsilon}_\Lambda = (\mathbf{\Psi}^T_{\Lambda_x}\boldsymbol{d_x}, \mathbf{\Psi}^T_{\Lambda_y}\boldsymbol{d_y}, \mathbf{\Psi}^T_{\Lambda_v}\boldsymbol{d_v}, (\mathbf{\Psi}^H_{\Lambda_w})^T\boldsymbol{d_w}) \in \Upsilon_\Lambda$ and $\boldsymbol{\lambda}_\Lambda = ((\mathbf{\Psi}^H_{\Lambda'_x})^T\boldsymbol{d_\mu}, \tilde{\mathbf{\Psi}}^T_{\Lambda_y}\boldsymbol{d_\nu}) \in \mathcal{M}_\Lambda$ into (6)–(7) and testing with all the basis functions in $\Upsilon_\Lambda$ 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. $\langle\langle \mathbf{\Psi}_\Lambda, \mathbf{\Psi}_{\Lambda'}\rangle\rangle = \left(\langle \psi_{(j,k)}, \psi_{(j',k')}\rangle\right)_{(j,k)\in\Lambda,(j',k')\in\Lambda'}$. Moreover, the Kronecker product of a matrix $\boldsymbol{A} \in I\!\!R^{n\times m}$ 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

$$\boldsymbol{Q}_\Lambda = \boldsymbol{Q} \otimes \langle\langle \boldsymbol{\Psi}_{\Lambda_y}, \boldsymbol{\Psi}_{\Lambda_y}\rangle\rangle, \quad \boldsymbol{R}_{\boldsymbol{v}_\Lambda} = \boldsymbol{R}_{\boldsymbol{v}} \otimes \langle\langle \boldsymbol{\Psi}_{\Lambda_v}, \boldsymbol{\Psi}_{\Lambda_v}\rangle\rangle, \quad \boldsymbol{R}_{\boldsymbol{w}_\Lambda} = \boldsymbol{R}_{\boldsymbol{w}} \otimes \boldsymbol{I}_{\Lambda_w}, \tag{10}$$

$$\boldsymbol{X}_\Lambda = \boldsymbol{I}_{n_x} \otimes \langle\langle \boldsymbol{\Psi}^H_{\Lambda'_x}, \dot{\boldsymbol{\Psi}}_{\Lambda_x}\rangle\rangle - \boldsymbol{A} \otimes \langle\langle \boldsymbol{\Psi}^H_{\Lambda'_x}, \boldsymbol{\Psi}_{\Lambda_x}\rangle\rangle, \ \boldsymbol{W}_\Lambda = \boldsymbol{W} \otimes \langle\langle \boldsymbol{\Psi}^H_{\Lambda'_x}, \boldsymbol{\Psi}^H_{\Lambda_w}\rangle\rangle, \tag{11}$$

$$\boldsymbol{I}_\Lambda = \boldsymbol{I}_{n_y} \otimes \boldsymbol{I}_{\Lambda_y}, \qquad \boldsymbol{C}_\Lambda = \boldsymbol{C} \otimes \langle\langle \tilde{\boldsymbol{\Psi}}_{\Lambda_y}, \boldsymbol{\Psi}_{\Lambda_x}\rangle\rangle, \qquad \boldsymbol{V}_\Lambda = \boldsymbol{V} \otimes \langle\langle \tilde{\boldsymbol{\Psi}}_{\Lambda_y}, \boldsymbol{\Psi}_{\Lambda_v}\rangle\rangle, \tag{12}$$

$$\boldsymbol{B}_\Lambda = \boldsymbol{B} \otimes \langle\langle \boldsymbol{\Psi}^H_{\Lambda'_x}, \boldsymbol{\Psi}^H_{\Lambda_u}\rangle\rangle \tag{13}$$

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 (6)–(7) to $(\Upsilon_\Lambda, \mathcal{M}_\Lambda)$ yields the KKT–system:

$$\begin{pmatrix} & & & & \boldsymbol{X}_\Lambda{}^T & -\boldsymbol{C}_\Lambda{}^T \\ & 2\boldsymbol{Q}_\Lambda & & & & \boldsymbol{I}_\Lambda \\ & & 2\boldsymbol{R}_{\boldsymbol{v}_\Lambda} & & & -\boldsymbol{V}^T_\Lambda \\ & & & 2\boldsymbol{R}_{\boldsymbol{w}_\Lambda} & -\boldsymbol{W}_\Lambda{}^T & \\ \boldsymbol{X}_\Lambda & & & -\boldsymbol{W}_\Lambda & & \\ -\boldsymbol{C}_\Lambda & \boldsymbol{I}_\Lambda & -\boldsymbol{V}_\Lambda & & & \end{pmatrix} \begin{pmatrix} \boldsymbol{d}_{\boldsymbol{x}} \\ \boldsymbol{d}_{\boldsymbol{y}} \\ \boldsymbol{d}_{\boldsymbol{v}} \\ \boldsymbol{d}_{\boldsymbol{w}} \\ \boldsymbol{d}_{\boldsymbol{\mu}} \\ \boldsymbol{d}_{\boldsymbol{\nu}} \end{pmatrix} = \begin{pmatrix} \boldsymbol{0} \\ 2\boldsymbol{Q}_\Lambda \boldsymbol{d}_{\boldsymbol{z}} \\ \boldsymbol{0} \\ \boldsymbol{0} \\ \boldsymbol{B}_\Lambda \boldsymbol{d}_{\boldsymbol{u}} \\ \boldsymbol{0} \end{pmatrix}. \tag{14}$$

This linear system is simply the Galerkin discretization of (8) 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

$$\min_{\boldsymbol{d}_{\boldsymbol{x}},\, \boldsymbol{d}_{\boldsymbol{y}},\, \boldsymbol{d}_{\boldsymbol{v}},\, \boldsymbol{d}_{\boldsymbol{w}}} \quad (\boldsymbol{d}_{\boldsymbol{y}} - \boldsymbol{d}_{\boldsymbol{z}})^T \boldsymbol{Q}_\Lambda (\boldsymbol{d}_{\boldsymbol{y}} - \boldsymbol{d}_{\boldsymbol{z}}) + \boldsymbol{d}_{\boldsymbol{v}}^T \boldsymbol{R}_{\boldsymbol{v}_\Lambda} \boldsymbol{d}_{\boldsymbol{v}} + \boldsymbol{d}_{\boldsymbol{w}}^T \boldsymbol{R}_{\boldsymbol{w}_\Lambda} \boldsymbol{d}_{\boldsymbol{w}}$$

subject to the constraints

$$\begin{aligned} \boldsymbol{X}_\Lambda \boldsymbol{d}_{\boldsymbol{x}} - \boldsymbol{W}_\Lambda \boldsymbol{d}_{\boldsymbol{w}} &= \boldsymbol{B}_\Lambda \boldsymbol{u}\,, \\ \boldsymbol{I}_\Lambda \boldsymbol{d}_{\boldsymbol{y}} - \boldsymbol{C}_\Lambda \boldsymbol{d}_{\boldsymbol{x}} - \boldsymbol{V}_\Lambda \boldsymbol{d}_{\boldsymbol{v}} &= \boldsymbol{0}\,, \end{aligned}$$

obtained by restricting the minimization in (1)–(3) to the finite dimensional space $\Upsilon_\Lambda$.

The linear system (14) has the following block structure

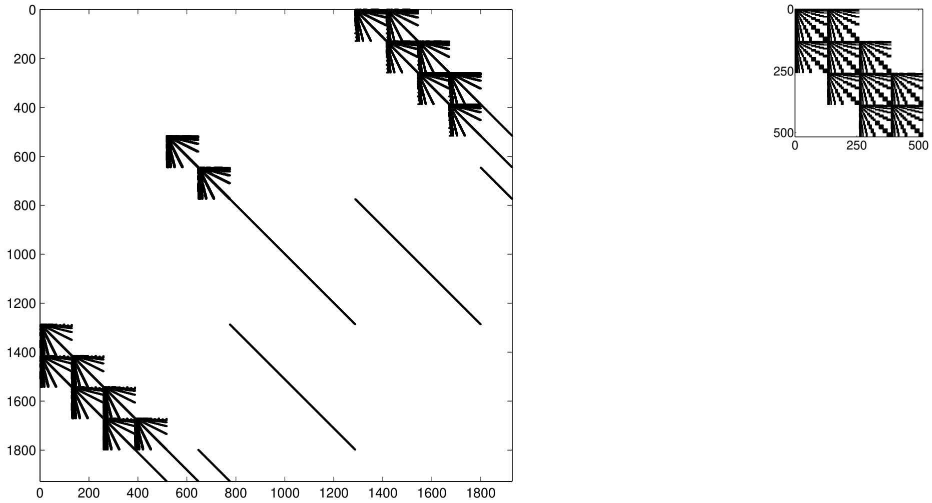
$$\boldsymbol{\mathcal{L}}_\Lambda \begin{pmatrix} \boldsymbol{\Upsilon}_\Lambda \\ \boldsymbol{\lambda}_\Lambda \end{pmatrix} := \begin{pmatrix} \boldsymbol{\mathcal{A}}_\Lambda & \boldsymbol{\mathcal{B}}_\Lambda{}^T \\ \boldsymbol{\mathcal{B}}_\Lambda & \boldsymbol{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{\Upsilon}_\Lambda \\ \boldsymbol{\lambda}_\Lambda \end{pmatrix} = \begin{pmatrix} \boldsymbol{h}_\Lambda \\ \mathrm{g}_\Lambda \end{pmatrix}, \tag{15}$$

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 1a. 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 $2^{-7}$.

# 4 Numerical Algorithm

In this section we want to outline the solution of the optimization problem (1) subject to the constraints (2)–(3), or equivalently, the corresponding operator equation (8). We employ

Figure 1: 1a. KKT–matrix $\boldsymbol{\mathcal{L}}_\Lambda$ 1b. Schur complement $\boldsymbol{\mathcal{S}}_\Lambda$ (30).




a refinement approach to provide a state estimation at any instant of time. Therefore, we consider a sequence of finite dimensional linear systems (15)

$$\boldsymbol{\mathcal{L}}_{\Lambda_\ell} \begin{pmatrix} \boldsymbol{\Upsilon}_{\Lambda_\ell} \\ \boldsymbol{\lambda}_{\Lambda_\ell} \end{pmatrix} = \begin{pmatrix} \boldsymbol{h}_{\Lambda_\ell} \\ \mathbf{g}_{\Lambda_\ell} \end{pmatrix} \qquad \text{for } \ell = \ell_0, \ell_0+1, \ldots \tag{16}$$

and apply to the whole sequence of equations a *nested iteration* scheme as introduced for example in (Ref. 21) (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 $\ell$ 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 (15) (Refs. 21,23,24). However, a direct application of the Uzawa technique is prohibited by the rank deficiency of the upper left block $\boldsymbol{\mathcal{A}}_\Lambda$ in (15). Nevertheless, due to the simplicity of the nonzero part of $\boldsymbol{\mathcal{A}}_\Lambda$, we can easily reduce system (14) 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:

$$\begin{pmatrix} \frac{1}{2}\left(\boldsymbol{Q}_\Lambda^{-1} + \boldsymbol{V}_\Lambda {\boldsymbol{R}_{\boldsymbol{v}_\Lambda}}^{-1} {\boldsymbol{V}_\Lambda}^T\right) & \boldsymbol{0} & \boldsymbol{C}_\Lambda \\ \boldsymbol{0} & \frac{1}{2}\boldsymbol{W}_\Lambda {\boldsymbol{R}_{\boldsymbol{w}_\Lambda}}^{-1} {\boldsymbol{W}_\Lambda}^T & -\boldsymbol{X}_\Lambda \\ {\boldsymbol{C}_\Lambda}^T & -{\boldsymbol{X}_\Lambda}^T & \boldsymbol{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{d_\nu} \\ \boldsymbol{d_\mu} \\ \boldsymbol{d_x} \end{pmatrix} = \begin{pmatrix} \boldsymbol{d_z} \\ -\boldsymbol{B}_\Lambda \boldsymbol{d_u} \\ \boldsymbol{0} \end{pmatrix} \tag{17}$$

and

$$\begin{aligned} 2\boldsymbol{Q}_\Lambda \boldsymbol{d_y} &= -\boldsymbol{d_\nu} + 2\boldsymbol{Q}_\Lambda \boldsymbol{d_z} \,, && (18) \\ 2\boldsymbol{R}_{\boldsymbol{v}_\Lambda} \boldsymbol{d_v} &= {\boldsymbol{V}_\Lambda}^T \boldsymbol{d_\nu} \,, && (19) \\ 2\boldsymbol{R}_{\boldsymbol{w}_\Lambda} \boldsymbol{d_w} &= {\boldsymbol{W}_\Lambda}^T \boldsymbol{d_\mu} \,. && (20) \end{aligned}$$

The equations (18)–(20) 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 (10)–(13) 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 (17) has again the structure (15)

$$\tilde{\boldsymbol{\mathcal{L}}}_\Lambda \begin{pmatrix} \tilde{\boldsymbol{\lambda}}_\Lambda \\ \tilde{\boldsymbol{\Upsilon}}_\Lambda \end{pmatrix} = \begin{pmatrix} \tilde{\boldsymbol{\mathcal{A}}}_\Lambda & \tilde{\boldsymbol{\mathcal{B}}}_\Lambda{}^T \\ \tilde{\boldsymbol{\mathcal{B}}}_\Lambda & \boldsymbol{0} \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{\lambda}}_\Lambda \\ \tilde{\boldsymbol{\Upsilon}}_\Lambda \end{pmatrix} = \begin{pmatrix} \tilde{\boldsymbol{h}}_\Lambda \\ \tilde{\mathbf{g}}_\Lambda \end{pmatrix}, \tag{21}$$

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

$$\tilde{\boldsymbol{\mathcal{L}}} \begin{pmatrix} \tilde{\boldsymbol{\lambda}} \\ \tilde{\boldsymbol{\Upsilon}} \end{pmatrix} = \begin{pmatrix} \tilde{\boldsymbol{h}} \\ \tilde{\mathbf{g}} \end{pmatrix}. \tag{22}$$

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 (14) 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 (21) to

$$\tilde{\boldsymbol{\mathcal{A}}}_\Lambda \tilde{\boldsymbol{\lambda}}_\Lambda = \tilde{\boldsymbol{h}}_\Lambda - \tilde{\boldsymbol{\mathcal{B}}}_\Lambda{}^T \tilde{\boldsymbol{\Upsilon}}_\Lambda \tag{23}$$

$$\tilde{\boldsymbol{\mathcal{B}}}_\Lambda \tilde{\boldsymbol{\mathcal{A}}}_\Lambda{}^{-1} \tilde{\boldsymbol{\mathcal{B}}}_\Lambda{}^T \tilde{\boldsymbol{\Upsilon}}_\Lambda = \tilde{\boldsymbol{\mathcal{B}}}_\Lambda \tilde{\boldsymbol{\mathcal{A}}}_\Lambda{}^{-1} \tilde{\boldsymbol{h}}_\Lambda - \tilde{\mathbf{g}}_\Lambda \tag{24}$$

by block elimination. The system (24) 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 (24) 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 (24) yields also in each step the search direction for solving (23). In this sense the Uzawa algorithm combines the solution of (24) and (23), avoiding an additional application of the pcg–method to (23).

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

$$\tilde{\boldsymbol{\mathcal{A}}}_\Lambda{}^{-1} = \begin{pmatrix} \left(\boldsymbol{Q}^{-1} + \boldsymbol{V}\boldsymbol{R_v}{}^{-1}\boldsymbol{V}^T\right)^{-1} \otimes \langle\langle \boldsymbol{\Psi}_{\Lambda_y}, \boldsymbol{\Psi}_{\Lambda_y}\rangle\rangle & \boldsymbol{0} \\ \boldsymbol{0} & (\boldsymbol{W}\boldsymbol{R_w}{}^{-1}\boldsymbol{W}^T)^{-1} \otimes \boldsymbol{I}_{\Lambda'_x} \end{pmatrix}. \tag{25}$$

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 (26). 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 (26) below. Thus we obtain the following algorithm for the system (21). To avoid a severe cluttering of indices we suppress here the index $\Lambda$ 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 $k = 0, 1, \dots$ compute:

$$\begin{aligned}
\boldsymbol{p}_k &= \boldsymbol{\mathcal{B}}^T\boldsymbol{d}_k\\
\boldsymbol{s}_k &= \boldsymbol{\mathcal{A}}^{-1}\boldsymbol{p}_k\\
\alpha_k &= \rho_k/(\boldsymbol{p}_k^T\boldsymbol{s}_k)\\
\boldsymbol{\Upsilon}_{k+1} &= \boldsymbol{\Upsilon}_k + \alpha_k\boldsymbol{d}_k\\
\boldsymbol{\lambda}_{k+1} &= \boldsymbol{\lambda}_k - \alpha_k\boldsymbol{s}_k\\
\boldsymbol{r}_{k+1} &= \mathrm{g} - \boldsymbol{\mathcal{B}}\boldsymbol{\lambda}_{k+1}\\
\boldsymbol{q}_{k+1} &= \boldsymbol{M}\boldsymbol{r}_{k+1}\\
\rho_{k+1} &= \boldsymbol{r}_{k+1}^T\boldsymbol{q}_{k+1}\\
\boldsymbol{d}_{k+1} &= -\boldsymbol{q}_{k+1} + \rho_{k+1}/\rho_k\ \boldsymbol{d}_k\,.
\end{aligned} \tag{26}$$

Note that this algorithm determines not only $\tilde{\boldsymbol{\Upsilon}}_\Lambda = \boldsymbol{d}_{\boldsymbol{x}}$ 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 Ref. 14), 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 $\kappa_2 := \mathrm{cond}_2(\boldsymbol{M}^{1/2}\boldsymbol{\mathcal{S}}_\Lambda\boldsymbol{M}^{1/2})$:

$$\|\ \boldsymbol{d}_{\boldsymbol{x}}^{(k)} - \boldsymbol{d}_{\boldsymbol{x}}\ \|_{\boldsymbol{\mathcal{S}}_\Lambda} \leq 2\ \|\ \boldsymbol{d}_{\boldsymbol{x}}^{(0)} - \boldsymbol{d}_{\boldsymbol{x}}\ \|_{\boldsymbol{\mathcal{S}}_\Lambda}\ \delta^k \tag{27}$$

where $\delta = \frac{\sqrt{\kappa_2}-1}{\sqrt{\kappa_2}+1}$ and where the *energy–norm* $\|\ \boldsymbol{d}_{\boldsymbol{x}}\ \|_{\boldsymbol{A}}^2$ is defined as $\boldsymbol{d}_{\boldsymbol{x}}^T\boldsymbol{A}\boldsymbol{d}_{\boldsymbol{x}}$ (Ref. 25).

As a consequence of the norm equivalence for wavelets (Ref. 14)

$$c_1\ \|\ \boldsymbol{d}_f\ \|_{\ell_2} \leq \|\ f\ \|_{L_2} \leq c_2\ \|\ \boldsymbol{d}_f\ \|_{\ell_2} \tag{28}$$

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 $\boldsymbol{x}_\Lambda = \boldsymbol{\Psi}_\Lambda^T \boldsymbol{d}_{\boldsymbol{x}}$ 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 $\bar{\boldsymbol{S}}_\Lambda$:

$$\| \boldsymbol{x}_\Lambda^{(k)} - \boldsymbol{x}_\Lambda \|_{\bar{\boldsymbol{S}}_\Lambda} \leq 2\bar{c} \, \| \boldsymbol{x}_\Lambda^{(0)} - \boldsymbol{x}_\Lambda \|_{\bar{\boldsymbol{S}}_\Lambda} \delta^k. \tag{29}$$

The constant $\bar{c} = \operatorname{cond}(\langle\langle \boldsymbol{\Psi}, \boldsymbol{\Psi} \rangle\rangle)$ is less than 12 for our choice of wavelets. The design of preconditioners that give rise to possibly small reduction factors $\delta$, 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 (14) to the unknowns $\boldsymbol{d}_{\boldsymbol{x}}$ gives the Schur complement

$$\boldsymbol{S}_\Lambda = 2 \left[ \boldsymbol{X}_\Lambda^T \left( \boldsymbol{W}_\Lambda \boldsymbol{R}_{\boldsymbol{w}_\Lambda}{}^{-1} \boldsymbol{W}_\Lambda^T \right)^{-1} \boldsymbol{X}_\Lambda + \boldsymbol{C}_\Lambda^T \left( \boldsymbol{Q}_\Lambda^{-1} + \boldsymbol{V}_\Lambda \boldsymbol{R}_{\boldsymbol{v}_\Lambda}{}^{-1} \boldsymbol{V}_\Lambda^T \right)^{-1} \boldsymbol{C}_\Lambda \right] \tag{30}$$

which is positive definite. The corresponding linear system is

$$1/2 \, \boldsymbol{S}_\Lambda \boldsymbol{d}_{\boldsymbol{x}} = \boldsymbol{X}_\Lambda^T \left( \boldsymbol{W}_\Lambda \boldsymbol{R}_{\boldsymbol{w}_\Lambda}{}^{-1} \boldsymbol{W}_\Lambda^T \right)^{-1} \boldsymbol{B}_\Lambda \boldsymbol{d}_{\boldsymbol{u}} + \boldsymbol{C}_\Lambda^T \left( \boldsymbol{Q}_\Lambda^{-1} + \boldsymbol{V}_\Lambda \boldsymbol{R}_{\boldsymbol{v}_\Lambda}{}^{-1} \boldsymbol{V}_\Lambda^T \right)^{-1} \boldsymbol{d}_{\boldsymbol{z}}. \tag{31}$$

Abbreviating $\bar{\boldsymbol{Q}} := (\boldsymbol{Q}^{-1} + \boldsymbol{V} \boldsymbol{R}_{\boldsymbol{v}}{}^{-1} \boldsymbol{V}^T)^{-1}$ and $\bar{\boldsymbol{R}}_{\boldsymbol{w}} := (\boldsymbol{W} \boldsymbol{R}_{\boldsymbol{w}}{}^{-1} \boldsymbol{W}^T)^{-1}$ and exploiting the rules for the Kronecker product we derive from (10)–(13) and (25) the following representation of $\boldsymbol{S}_\Lambda$

$$\begin{aligned} 1/2 \, \boldsymbol{S}_\Lambda \;=\; & \bar{\boldsymbol{R}}_{\boldsymbol{w}} \otimes \left( \langle\langle \dot{\boldsymbol{\Psi}}_{\Lambda_x}, \boldsymbol{\Psi}_{\Lambda_x'}^H \rangle\rangle \langle\langle \boldsymbol{\Psi}_{\Lambda_x'}^H, \dot{\boldsymbol{\Psi}}_{\Lambda_x} \rangle\rangle \right) \\ & - \left\{ \left( \boldsymbol{A}^T \bar{\boldsymbol{R}}_{\boldsymbol{w}} \right) \otimes \left( \langle\langle \boldsymbol{\Psi}_{\Lambda_x}, \boldsymbol{\Psi}_{\Lambda_x'}^H \rangle\rangle \langle\langle \boldsymbol{\Psi}_{\Lambda_x'}^H, \dot{\boldsymbol{\Psi}}_{\Lambda_x} \rangle\rangle \right) \right\} \\ & - \left\{ \left( \boldsymbol{A}^T \bar{\boldsymbol{R}}_{\boldsymbol{w}} \right) \otimes \left( \langle\langle \boldsymbol{\Psi}_{\Lambda_x}, \boldsymbol{\Psi}_{\Lambda_x'}^H \rangle\rangle \langle\langle \boldsymbol{\Psi}_{\Lambda_x'}^H, \dot{\boldsymbol{\Psi}}_{\Lambda_x} \rangle\rangle \right) \right\}^T \\ & + \left( \boldsymbol{A}^T \bar{\boldsymbol{R}}_{\boldsymbol{w}} \boldsymbol{A} \right) \otimes \left( \langle\langle \boldsymbol{\Psi}_{\Lambda_x}, \boldsymbol{\Psi}_{\Lambda_x'}^H \rangle\rangle \langle\langle \boldsymbol{\Psi}_{\Lambda_x'}^H, \boldsymbol{\Psi}_{\Lambda_x} \rangle\rangle \right) \\ & + \left( \boldsymbol{C}^T \bar{\boldsymbol{Q}} \boldsymbol{C} \right) \otimes \left( \langle\langle \boldsymbol{\Psi}_{\Lambda_x}, \tilde{\boldsymbol{\Psi}}_{\Lambda_y} \rangle\rangle \langle\langle \boldsymbol{\Psi}_{\Lambda_y}, \boldsymbol{\Psi}_{\Lambda_y} \rangle\rangle \langle\langle \tilde{\boldsymbol{\Psi}}_{\Lambda_y}, \boldsymbol{\Psi}_{\Lambda_x} \rangle\rangle \right) . \end{aligned} \tag{32}$$

Note that all the wavelet matrices appearing in (32) 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 1b. exhibits the nonzero entries of $\boldsymbol{S}_\Lambda$ corresponding to the KKT–system in Figure 1a. 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 (8) and its finite dimensional approximation $\boldsymbol{\mathcal{L}}_\Lambda$ in (15). 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 $\Lambda$.

Hence, system (14) 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 (9) $\boldsymbol{\mathcal{L}}$ 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 (33), (34)) 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 (Ref. 10).

To this end, recall from (Ref. 20) that the piecewise linear wavelet bases used in Section 3 for the discretization of the states induces the following isomorphism between $\ell_2$ 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

$$c_3\|\{2^j d_{j,k}\}_{j,k}\|_{\ell_2} \leq \left(\|f\|^2_{L_2} + \|\dot{f}\|^2_{L_2}\right)^{1/2} = \|f\|_{H^1} \leq c_4\|\{2^j d_{j,k}\}_{j,k}\|_{\ell_2}. \tag{33}$$

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

$$c_5\|\{2^{-j}\tilde{d}_{j,k}\}_{j,k}\|_{\ell_2} \leq \|f\|_{(H^1)'} \leq c_6\|\{2^{-j}\tilde{d}_{j,k}\}_{j,k}\|_{\ell_2}. \tag{34}$$

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 Refs. 14,26 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

$$\mathrm{cond}_2\left(\boldsymbol{D}_\Lambda^{-1}\boldsymbol{\mathcal{S}}_\Lambda\boldsymbol{D}_\Lambda^{-1}\right) = \mathcal{O}(1) \tag{35}$$

independently of $\Lambda$.

*Proof:* The operator $\tilde{\boldsymbol{\mathcal{L}}}$ (see (22)) 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 (9) that there exist positive constants $c_7, c_8$ for $\tilde{\boldsymbol{\mathcal{L}}}$ such that

$$\begin{aligned} c_7\left(\|\tilde{\boldsymbol{\lambda}}\|^2_{(L_2^{n_y}\times L_2^{n_w})} + \|\boldsymbol{x}\|^2_{(H^1)^{n_x})}\right)^{1/2} &\leq \|\tilde{\boldsymbol{\mathcal{L}}}\begin{pmatrix}\tilde{\boldsymbol{\lambda}}\\ \boldsymbol{x}\end{pmatrix}\|_{(L_2^{n_y}\times L_2^{n_w})\times((H^1)^{n_x})'} \\ &\leq c_8\left(\|\tilde{\boldsymbol{\lambda}}\|^2_{(L_2^{n_y}\times L_2^{n_w})} + \|\boldsymbol{x}\|^2_{(H^1)^{n_x})}\right)^{1/2}. \end{aligned} \tag{36}$$

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

(28) for the first two components and employing (33) for the third component, the latter scaling of $\boldsymbol{\Psi}$ just ensures that

$$c_3\left(\|\boldsymbol{d}_{\boldsymbol{\nu}}\|_{\ell_2}^2+\|\boldsymbol{d}_{\boldsymbol{\mu}}\|_{\ell_2}^2+\|\hat{\boldsymbol{d}}_{\boldsymbol{x}}\|_{\ell_2}^2\right)^{1/2}\le\left(\|\tilde{\boldsymbol{\lambda}}\|_{L_2}^2+\|\boldsymbol{x}\|_{H^1}^2\right)^{1/2}$$
$$\le c_4\left(\|\boldsymbol{d}_{\boldsymbol{\nu}}\|_{\ell_2}^2+\|\boldsymbol{d}_{\boldsymbol{\mu}}\|_{\ell_2}^2+\|\hat{\boldsymbol{d}}_{\boldsymbol{x}}\|_{\ell_2}^2\right)^{1/2}. \tag{37}$$

Moreover, since the norm equivalences (28) and (34) give rise to

$$1/c_9\left\|\tilde{\mathcal{L}}\begin{pmatrix}\tilde{\boldsymbol{\lambda}}\\ \boldsymbol{x}\end{pmatrix}\right\|_{L_2^{ny}\times L_2^{nw}\times((H^1)^{nx})'}\le\left\|\begin{pmatrix}\tilde{\boldsymbol{\mathcal{A}}} & \tilde{\boldsymbol{\mathcal{B}}}^T\boldsymbol{D}^{-1}\\ \boldsymbol{D}^{-1}\tilde{\boldsymbol{\mathcal{B}}} & \boldsymbol{0}\end{pmatrix}\begin{pmatrix}(\boldsymbol{d}_{\boldsymbol{\nu}}^T,\boldsymbol{d}_{\boldsymbol{\mu}}^T)^T\\ \hat{\boldsymbol{d}}_{\boldsymbol{x}}\end{pmatrix}\right\|_{\ell_2}$$
$$\le 1/c_{10}\left\|\tilde{\mathcal{L}}\begin{pmatrix}\tilde{\boldsymbol{\lambda}}\\ \boldsymbol{x}\end{pmatrix}\right\|_{L_2^{ny}\times L_2^{nw}\times((H^1)^{nx})'} \tag{38}$$

with $c_9=\max\{c_2,c_6\}$ and $c_{10}=\min\{c_1,c_5\}$, we conclude from (36), (37) and (38) that

$$c_3c_7/c_9\left(\|\boldsymbol{d}_{\boldsymbol{\nu}}\|_{\ell_2}^2+\|\boldsymbol{d}_{\boldsymbol{\mu}}\|_{\ell_2}^2+\|\hat{\boldsymbol{d}}_{\boldsymbol{x}}\|_{\ell_2}^2\right)^{1/2}\le\left\|\begin{pmatrix}\tilde{\boldsymbol{\mathcal{A}}} & \tilde{\boldsymbol{\mathcal{B}}}^T\boldsymbol{D}^{-1}\\ \boldsymbol{D}^{-1}\tilde{\boldsymbol{\mathcal{B}}} & \boldsymbol{0}\end{pmatrix}\begin{pmatrix}(\boldsymbol{d}_{\boldsymbol{\nu}}^T,\boldsymbol{d}_{\boldsymbol{\mu}}^T)^T\\ \hat{\boldsymbol{d}}_{\boldsymbol{x}}\end{pmatrix}\right\|_{\ell_2}$$
$$\le c_4c_8/c_{10}\left(\|\boldsymbol{d}_{\boldsymbol{\nu}}\|_{\ell_2}^2+\|\boldsymbol{d}_{\boldsymbol{\mu}}\|_{\ell_2}^2+\|\hat{\boldsymbol{d}}_{\boldsymbol{x}}\|_{\ell_2}^2\right)^{1/2}. \tag{39}$$

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 (39) 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 $\Lambda$ 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 $\Lambda$ 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 (Refs. 21,23). 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 (33) 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 (28), (33) and (34) 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 (25)) 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

$$\begin{aligned} \left(\boldsymbol{M}_\Lambda^{aJ}\right)_{i,i'} &= \Bigg(\sum_s \left(\boldsymbol{Q}_\Lambda^{-1} + \boldsymbol{V}_\Lambda \boldsymbol{R}_{\boldsymbol{v}_\Lambda}{}^{-1} \boldsymbol{V}_\Lambda^T\right)^{-1}_{(s,s)} \boldsymbol{C}^2_{\Lambda,(s,i)} + \\ &\qquad \sum_s \left(\boldsymbol{W}_\Lambda \boldsymbol{R}_{\boldsymbol{w}_\Lambda}{}^{-1} \boldsymbol{W}_\Lambda^T\right)^{-1}_{(s,s)} \boldsymbol{X}^2_{\Lambda,(s,i)}\Bigg)^{-1} \delta_{i,i'}\,. \end{aligned} \tag{40}$$

Recall from (25) that the computation of the above diagonal matrices simply amounts to replacing the wavelet finger block matrices in (25) 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 Ref. 10 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 (Ref. 21) 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* $\alpha$ 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 (27), the maximal number of iterations $max_{it}$ on each level needed to realize the error reduction rate $\alpha$ 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 (35)), the maximal number of iterations on each level $max_{it}$ required by a fixed error reduction

$$||\boldsymbol{x} - \boldsymbol{x}_{\Lambda_\ell}||_{\mathcal{S}} \leq \alpha ||\boldsymbol{x} - \boldsymbol{x}_{\Lambda_{\ell-1}}||_{\mathcal{S}} \tag{41}$$

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 (16), respectively the $k$th 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 (29) that it suffices to choose the number of iterations $max_{it}$ so as to guarantee that

$$||\tilde{\boldsymbol{\Upsilon}}_{\Lambda_\ell} - \tilde{\boldsymbol{\Upsilon}}_{\Lambda_\ell}^{(k)}||_{\boldsymbol{\mathcal{S}}_{\Lambda_\ell}} \leq \alpha/\bar{c}\; ||\tilde{\boldsymbol{\Upsilon}}_{\Lambda_\ell} - \tilde{\boldsymbol{\Upsilon}}_{\Lambda_\ell}^{(0)}||_{\boldsymbol{\mathcal{S}}_{\Lambda_\ell}}, \tag{42}$$

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 (42). First, we can conclude from (29) and (42) that at most

$$max_{it} \approx log\,(2\bar{c}/\alpha)\,/\,(-log\,((\sqrt{\kappa_2}-1)/(\sqrt{\kappa_2}+1))) \tag{43}$$

iterations are needed. The second possibility is to use the estimation by the $\ell_2$–norm of the residuals of the preconditioned system

$$||\tilde{\boldsymbol{\Upsilon}}_{\Lambda_\ell} - \tilde{\boldsymbol{\Upsilon}}_{\Lambda_\ell}^{(k)}||_{\boldsymbol{\mathcal{S}}_{\boldsymbol{\Lambda_\ell}}} / ||\tilde{\boldsymbol{\Upsilon}}_{\Lambda_\ell} - \tilde{\boldsymbol{\Upsilon}}_{\Lambda_\ell}^{(0)}||_{\boldsymbol{\mathcal{S}}_{\boldsymbol{\Lambda_\ell}}} \leq \sqrt{\kappa_2}\;||\boldsymbol{M}_\Lambda^{1/2} res_{\Lambda_\ell}^{(k)}||_{\ell_2} / ||\boldsymbol{M}_\Lambda^{1/2} res_{\Lambda_\ell}^{(0)}||_{\ell_2} \tag{44}$$

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

$$||\boldsymbol{M}_\Lambda^{1/2} res_{\Lambda_\ell}^{(k)}||_{\ell_2} / ||\boldsymbol{M}_\Lambda^{1/2} res_{\Lambda_\ell}^{(0)}||_{\ell_2} \leq \alpha/(\bar{c}\sqrt{\kappa_2}) =: \alpha_{rel}\,. \tag{45}$$

In general the exact error reduction factor $\alpha$ is not computable. However, in our particular situation $\alpha$ 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 $\alpha$ for scalewise refinement to $\alpha = 1/2$. Moreover, to determine $\alpha_{rel}$ and $max_{it}$ an upper bound for $\kappa_2$ has to be estimated. Choosing $\alpha_{rel} = 10^{-3}$ for all $j$ would cover condition numbers $\kappa_2$ 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 (Refs. 27–28). 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 (21) can then be summarized as follows:

set $\ell := \ell_0$ and solve (21) 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 $\tilde{\boldsymbol{\Upsilon}}_{\Lambda_\ell}^{(0)} \longleftarrow \tilde{\boldsymbol{\Upsilon}}_{\Lambda_{\ell-1}}^{(k)}$;
- apply the approximate Jacobi preconditioner $\boldsymbol{M}_{\Lambda_\ell}^{aJ}$;
- iterate the simplified Uzawa algorithm ($k := k+1$) until
  a.) the relative residual error in the $\ell_2$–norm is below $\alpha_{rel}$ 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 (Ref. 10). We choose $n_x = n_y = 4$. Furthermore, we allow model uncertainties in all state equations (2), i.e. $\boldsymbol{W} = \boldsymbol{I}$, but not in the output equations (3). 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 (2) and (3) with $\boldsymbol{w} = 0$, $\boldsymbol{x}(0) = \boldsymbol{1}$ 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 (43) obtained for $\alpha = 1/2$ and $\boldsymbol{M}_\Lambda = \boldsymbol{M}_\Lambda^{aJ}$. Obviously the condition numbers $cond(\boldsymbol{\mathcal{S}}_\Lambda)$ increase roughly

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

| Scale | $\boldsymbol{M}_\Lambda = \boldsymbol{I}$ | $\boldsymbol{M}_\Lambda = \boldsymbol{M}_\Lambda^{aJ}$ | $max_{it}$ |
|---|---|---|---|
| 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 |

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 $\ell_2$–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 (31). The last line gives the results for solving the system only on scale 9 with $\boldsymbol{0}$ as initial guess in the simplified Uzawa algorithm.

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

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 |

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 (15) 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 (Ref. 6). On the other hand, level depending regularization parameters can be incorporated. For example, in (Ref. 29) 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 (Ref. 30).

# References


[1] MICHALSKA, H., and MAYNE, D.Q., *Moving Horizon Observers,* IFAC Symposium NOLCOS'92, 1992.

[2] MUSKE, K.R., and RAWLINGS, J.B., *Nonlinear Receding Horizon State Estimation,* Methods of Model–Based Control, Edited by Berber, R., NATO ASI Series, Kluwer Press, Dordrecht, the Netherlands, 1995.

[3] ROBERTSON, D., and LEE, J.H., and RAWLINGS, J.B., *A Moving Horizon–Based Approach for Least–Squares Estimation,* AIChE Journal, Vol.42, pp.2209-2223, 1996.

[4] BINDER, T., and BLANK, L., and DAHMEN, W., and MARQUARDT, W., *Towards Multiscale Dynamic and Data Reconciliation,* Nonlinear Model Based Process Control, Edited by Berber, R., and Kravaris, C., NATO ASI Series, Kluwer Academic Publishers, Dordrecht, the Netherlands, pp.623-665, 1998.

[5] HIRSCHHORN, R.M., *Invertability of Multivariable Nonlinear Control Systems,* IEEE Transactions on Automatic Control, Vol.24, pp.855-865, 1979.

[6] BINDER, T., and BLANK, L., and DAHMEN, W., and MARQUARDT, W., *On the Regularization of Dynamic Data Reconciliation Problems,* Journal of Process Control, (to appear), 2001.

[7] TIKHONOV, A., and ARSENIN, V., *Solutions of Ill–posed Problems,* Wiley, New York, New York, 1977.

[8] BIEGLER, L.T., *Efficient Solution of Dynamic Optimization and NMPC Problems,* Nonlinear Model Predictive Control, Edited by Allgöwer, F., and Zheng, A., Birkhäuser Verlag, Basel, Switzerland, pp.219-243, 2000.

[9] BOCK, H.G., and DIEHL, M.M., and LEINEWEBER, D.B., and SCHLÖDER, J.P., *A Direct Multiple Shooting Method for Real–time Optimization of Nonlinear DAE Processes,* Nonlinear Model Predictive Control, Edited by Allgöwer, F., and Zheng, A., Birkhäuser Verlag, Basel, Switzerland, pp.245-267, 2000.

[10] BINDER, T., and BLANK, L., and DAHMEN, W., and MARQUARDT, W., *Iterative Algorithms for Multiscale State Estimation, Part2: Numerical Investigation,* Journal of Optimization Theory and Applications (to appear), 2001.

[11] BARD, J., *Nonlinear Parameter Estimation,* Academic Press, New York, New York, 1974.

[12] KAILATH, T., *Linear Systems,* Prentice Hall, Englewood Cliffs, New Jersey, 1980.

[13] COHEN, A., and MASSON, R., *Adaptive Wavelet Methods for Second Order Elliptic Problems, Preconditioning and Adaptivity,* SIAM Journal on Scientific Computing, Vol.21, pp.1006-1026, 1999.

[14] DAHMEN, W., *Wavelet and Multiscale Methods for Operator Equations,* Acta Numerica, Vol.7, pp.55-228, 1997.

[15] CHENG, Z., and PILKEY, W.D., *Wavelet–Based Limiting Performance Analysis of Mechanical Systems subject to Transient Disturbances,* Finite Elements in Analysis and Design, Vol. 33, pp.233-245, 1999.

[16] HSIAO, C.H., and WANG, W.J., *Optimal Control of Time–Varying Systems via Haar Wavelets,* Journal of Optimization Theory and Applications, Vol. 103, pp.641-655, 1999.

[17] ZHOU, D., and CAI, W., and ZHANG, W., *An Adaptive Wavelet Method for Nonlinear Circuit Simulation,* IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, Vol.46, pp.931-938, 1999.

[18] CHUI, C.K., *An Introduction to Wavelets,* Academic Press, Boston, Massachusetts, 1992.

[19] DAUBECHIES, I., *Ten Lectures on Wavelets,* SIAM, Philadelphia, Pennsylvania 1992.

[20] DAHMEN, W., and KUNOTH, A., and URBAN, K., *Biorthogonal Spline–Wavelets on the Interval – Stability and Moment Conditions,* Applied and Computational Harmonic Analysis, Vol.6, pp.132-196, 1999.

[21] BRAESS, D., *Finite Elemente,* Springer Verlag, Berlin, Germany, 1992.

[22] BINDER, T., and BLANK, L., and DAHMEN, W., and MARQUARDT, W., *An Adaptive Multiscale Method for Real–Time Moving Horizon Optimization,* Proceedings of the American Control Conference, Chicago, Illinois, Omnipress, Madison, Wisconsin, pp.4234-4238, 2000.

[23] BREZZI, F., and FORTIN, M., *Mixed and Hybrid Finite Element Methods,* Springer Verlag, Berlin, Germany, 1991.

[24] GLOWINSKI, R., *Numerical Methods for Nonlinear Variational Problems,* Springer, New York, New York, 1984.

[25] GOLUB, G.H., and VAN LOAN, C.F., *Matrix Computations,* John Hopkins Press, London, Great Britain, 1996.

[26] DAHMEN, W., and KUNOTH, A., *Multilevel Preconditioning,* Numerische Mathematik, Vol.63, pp.315-344, 1992.

[27] AXELSSON, O., *Iterative Solution Methods,* Cambridge University Press, New York, New York, 1994.

[28] MEURANT, G., *Computer Solution of Large Linear Systems,* Elsevier Science, Amsterdam, the Netherlands, 1999.

[29] MAAß, P., and RIEDER, A., *Wavelet–Accelerated Tikhonov–Philipps Regularization with Applications,* Inverse Problems in Medical Imaging and Nondestructive Testing, Edited by H.W. Engl, A.K. Louis, W. Rundell, Springer, Wien, Austria, pp.134–158, 1997.

[30] MORARI, M., and LEE, J.H., *Model Predictive Control: Past, Present and Future,* Computers & Chemical Engineering, Vol.23, pp.667-682, 1999.