# Multiscale Concepts for Moving Horizon Optimization

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

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

**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.

**Keywords**: 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 [2] in this book and references therein. Both, the regulator problem and the estimation problem have to be solved repetitively within a fixed time span $\Delta T$ which is dictated by the dynamics of the process. An acceptable algorithm has to prompt the optimal values within $\Delta T$ since proper process operation cannot be guaranteed otherwise.

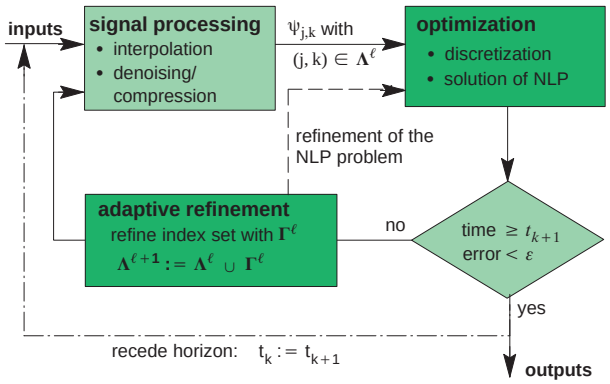
The majority of known implementations of dynamic optimization algorithms for large–scale systems are based on the direct approach [2] since it does not require any analytic expression for the necessary optimality conditions which can be quite cumbersome to determine especially in the presence

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) [18] or by parameterizing the control variables only (sequential approach) [38]. 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. [12,14,38], 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 $\Delta T$ 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 $\Delta T$ 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 $\Delta T$ 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 when progressing to the next finer approximation. Hence, in the very spirit of classical *nested iteration* [13], 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 [11].

Figure 1 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 ini-

**Fig. 1.** Receding Horizon Estimation.




tial mesh, respectively a first set of basis functions $\Lambda^0$, 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 $\ell$ 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 $\Delta T$. 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 [15,16,24]. 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 [16,17,19,21].

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 [7,8]. 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 [29]. 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 $\boldsymbol{x} \in \left(H^1\right)^{n_x}$, $\boldsymbol{y} \in (L_2)^{n_y}$, $\boldsymbol{v} \in (L_2)^{n_v}$ 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

$$\min_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{v},\boldsymbol{w}} \int_{t_{k-m}}^{t_k} \left\{ (\boldsymbol{y}-\boldsymbol{z})^T \boldsymbol{Q} (\boldsymbol{y}-\boldsymbol{z}) \right\} \, d\tau \tag{1}$$

subject to the constraints

$$\dot{\boldsymbol{x}} - \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}) - \boldsymbol{W}\boldsymbol{w} = \boldsymbol{0} \ , \tag{2}$$

$$\boldsymbol{y} - \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}) - \boldsymbol{V}\boldsymbol{v} = \boldsymbol{0} \ , \tag{3}$$

$$\boldsymbol{l}_{\boldsymbol{x}}(t) \leq \boldsymbol{x}(t) \leq \boldsymbol{s}_{\boldsymbol{x}}(t) \ , \tag{4}$$

$$\boldsymbol{l}_{\boldsymbol{y}}(t) \leq \boldsymbol{y}(t) \leq \boldsymbol{s}_{\boldsymbol{y}}(t) \ , \tag{5}$$

$$t \in [t_{k-m}, t_k] \quad .$$

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 (1)–(5) is to be solved to determine the controls $\boldsymbol{u}$.

## 2 Problem Regularization

The goal of the problem formulation (1)–(5) 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 (2)–(3). For a general discussion of necessary and sufficient conditions concerning the invertibility of the process model we refer the reader to [28,36]. Even if a unique optimum exists, the inverse problem (1)–(5) 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 [25,30] 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 (2), (3) this may cause $\boldsymbol{x}$ to have large derivatives which gives rise to arbitrarily large variation of $\boldsymbol{w}$ in $L_2$ [7]. 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 (1)–(5) 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 [37]. Then, the extended problem formulation

$$\min_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{v},\boldsymbol{w}} \int_{t_k-m}^{t_k} \left\{ (\boldsymbol{y}-\boldsymbol{z})^T \boldsymbol{Q} (\boldsymbol{y}-\boldsymbol{z}) + \boldsymbol{v}^T \boldsymbol{R}_{\boldsymbol{v}} \boldsymbol{v} + \boldsymbol{w}^T \boldsymbol{R}_{\boldsymbol{w}} \boldsymbol{w} \right\} \, d\tau \tag{6}$$

with constraints (2)–(5) 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 (1)–(5) into finite dimensional subspaces, e.g. using a Petrov–Galerkin scheme [34]. 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 [25,30]. Of course, regularization by projection might be combined with direct regularization methods based on penalty terms such as in (6).

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 [32]. However, since this information is not available for most practical problems we examined in [7] a strategy, namely the L–curve criterion [27], 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 [7] we explore the performance of regularization by projection of (1)–(5) 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 [7] 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* $\psi$, 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 $\psi$ is chosen to have bounded support, as it will always be the case in our applications, one has $diam\, supp\, \psi_{j,k} \sim 2^{-j}$, $j \to \infty$. The key feature is that for suitable $\psi$ 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

$$c_1 \parallel \boldsymbol{d}_f \parallel_{\ell_2} \leq \parallel f \parallel_{L_2} \leq c_2 \parallel \boldsymbol{d}_f \parallel_{\ell_2} \tag{7}$$

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, (7) implies the existence of a *dual basis* $\tilde{\boldsymbol{\Psi}}$ in $L_2$ which is *biorthogonal* to $\boldsymbol{\Psi}$, i.e.

$$\langle\langle \boldsymbol{\Psi}, \tilde{\boldsymbol{\Psi}} \rangle\rangle = \boldsymbol{I} \tag{8}$$

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, (7) 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, $S_j = \text{span}\,\{\varphi_{j,k} : k \in \mathcal{I}_j\}$, which are obtained by a suitable, compactly supported function $\varphi$. Due to the compact support of $\varphi$ 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 $W_j = \text{span}\,\{\psi_{j,k} : k \in \mathcal{I}'_j\} \perp \tilde{S}_j$. 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 (8).

All properties above can be realized when choosing $\varphi$ 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*

$$\langle (\cdot)^r, \psi \rangle = 0, \qquad r = 0, \ldots, \tilde{m} - 1 \; . \tag{9}$$

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 (2), (3) in a weak sense. We obtain the equality constraints:

$$\langle \dot{\boldsymbol{x}} - \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}) - \boldsymbol{W}\boldsymbol{w}, \boldsymbol{\nu}_1 \rangle = \boldsymbol{0} \qquad \text{for all } \boldsymbol{\nu}_1 \in L_2^{n_x} \tag{10}$$

$$\langle \boldsymbol{y} - \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{p}) - \boldsymbol{V}\boldsymbol{v}, \boldsymbol{\nu}_2 \rangle = \boldsymbol{0} \qquad \text{for all } \boldsymbol{\nu}_2 \in L_2^{n_y} \; . \tag{11}$$

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

$$\min_{\boldsymbol{\zeta}_\Lambda} \ \boldsymbol{\zeta}_\Lambda^T \boldsymbol{H}_\Lambda \boldsymbol{\zeta}_\Lambda + \boldsymbol{b}_\Lambda^T \boldsymbol{\zeta}_\Lambda \tag{12}$$

$$\text{s.t.} \qquad \boldsymbol{F}_\Lambda(\boldsymbol{\zeta}_\Lambda) = \boldsymbol{0} \tag{13}$$

$$\boldsymbol{n}_\Lambda(\boldsymbol{\zeta}_\Lambda) \leq \boldsymbol{0} \ . \tag{14}$$

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 $\Lambda_{\boldsymbol{x}}^0$ etc. and *adaptively* enlarge these index sets to $\Lambda_{\boldsymbol{x}}^{\ell+1} = \Lambda_{\boldsymbol{x}}^\ell \cup \Gamma_{\boldsymbol{x}}$ 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 [15,22–24]. 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 [15]. 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 $z^\delta = \boldsymbol{\Psi}^T \boldsymbol{d}_z^\delta$. 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 $z = \boldsymbol{\Psi}^T \boldsymbol{d}_z$ 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 $\varepsilon$ depends only on the noise level. If little or nothing is known about the noise, $\varepsilon$ 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 [10].

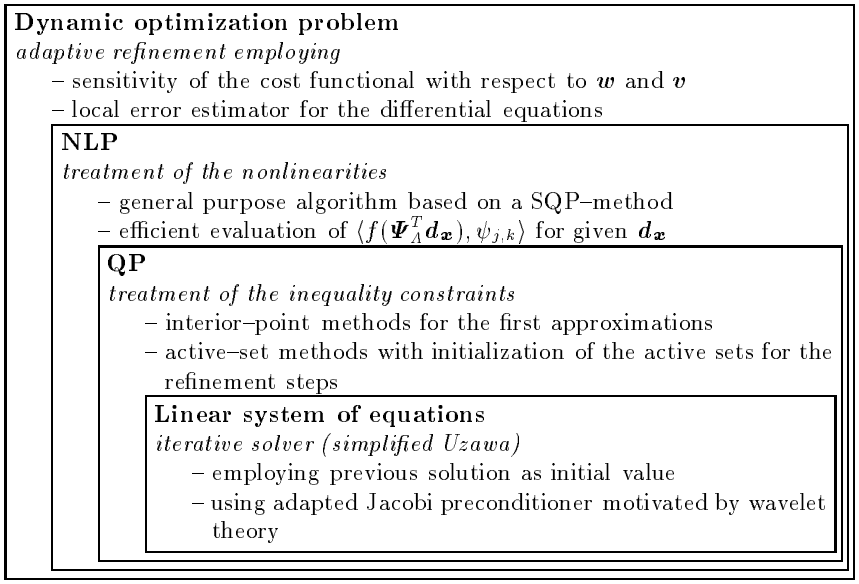
The representation of the measurements in terms of wavelets is fully compatible with our multiscale discretization of the constrained optimization problem. In particular, since the best $N$–term wavelet approximation is essentially determined by the first $N$ largest (in modulus) wavelet coefficients, the initial index set $\Lambda_0$ 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 2 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 2. 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

**Fig. 2.** Outline of the conceptual blocks on a fixed horizon.




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.

### 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 $\Lambda^0$. Then, based on the information of the previously computed approximation $\boldsymbol{d}_\Lambda^{*\ell}$ at refinement step $\ell$ an improved index set $\Lambda^{\ell+1}$ is generated and (12)–(14) is resolved. Algorithmic details on the adaptive refinement strategy are given in [5,9]. 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 $z_\Lambda^\ell$ satisfying

$$\| z - z_\Lambda^\ell \|_{L_2} \leq \varepsilon'_\ell \| z \|_{L_2} \tag{15}$$

with given tolerance $\varepsilon'_\ell$. 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 $\varepsilon$ which uniquely depends on $\varepsilon'_\ell$. The remaining

entries are the *significant* coefficients and their indices form the index set $\Lambda_z^\ell$. 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 [29]. 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 $\Lambda_x^\ell$. 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 $\Gamma^\ell$ the set of newly added indices:

1. Solve problem (12)–(14) with the index set $\Lambda^\ell$.
2. Refine $\boldsymbol{z}, \boldsymbol{u}$ by *thresholding* $\rightarrow \Gamma_z^\ell, \Gamma_u^\ell$.
3. Apply a *local error* estimator for $\boldsymbol{x}$, $\boldsymbol{y} \longrightarrow \Gamma_{x_i}^\ell, \Gamma_{y_i}^\ell$.
4. Use the *sensitivity* of Lagrange functional for $\boldsymbol{v}$, $\boldsymbol{w} \longrightarrow \Gamma_{v_i}^\ell, \Gamma_{w_i}^\ell$.
5. *Compress* current approximation by thresholding $\longrightarrow \tilde{\Lambda}_{x_i}^\ell$ 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.

**Fig. 3.** Scalewise versus adaptive refinement.

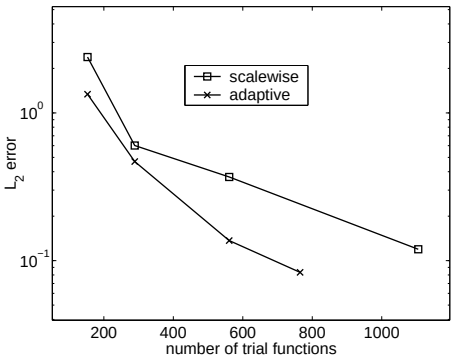



Figure 3 illustrates for a typical example the comparison between different choices of the index sets $\Lambda^\ell$, i.e. a uniform refinement which adds all basis functions of the next scale and an adaptive refinement which is based on the considerations above. The $L_2$ error is plotted versus the 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 (12)–(14) 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 [35]. 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 [26] typically requires routines to evaluate the cost functional in (12) and the residuals of the restrictions (13), (14) 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 (13), (14) 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 $\Lambda$. Then, in each refinement step $\ell$ those rows and

columns of the precomputed and stored inner product matrix whose indices are contained in $\Lambda^\ell$ 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.* [20] 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 $\ell$. 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 (14). 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 [40]. 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 [40]. 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 [6,39]. 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:

$$\boldsymbol{\mathcal{L}}_\Lambda \begin{pmatrix} \boldsymbol{\Upsilon}_\Lambda \\ \boldsymbol{\lambda}_\Lambda \end{pmatrix} := \begin{pmatrix} \boldsymbol{\mathcal{A}}_\Lambda & \boldsymbol{\mathcal{B}}_\Lambda{}^T \\ \boldsymbol{\mathcal{B}}_\Lambda & \mathbf{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{\Upsilon}_\Lambda \\ \boldsymbol{\lambda}_\Lambda \end{pmatrix} = \begin{pmatrix} \boldsymbol{h}_\Lambda \\ \mathrm{g}_\Lambda \end{pmatrix} , \tag{16}$$

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

$$\min_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{v},\boldsymbol{w}} \int_0^1 \left\{ (\boldsymbol{y}-\boldsymbol{z})^T \boldsymbol{Q} (\boldsymbol{y}-\boldsymbol{z}) \; + \; \boldsymbol{v}^T \boldsymbol{R}_{\boldsymbol{v}} \boldsymbol{v} + \boldsymbol{w}^T \boldsymbol{R}_{\boldsymbol{w}} \boldsymbol{w} \right\} \, d\tau \tag{17}$$

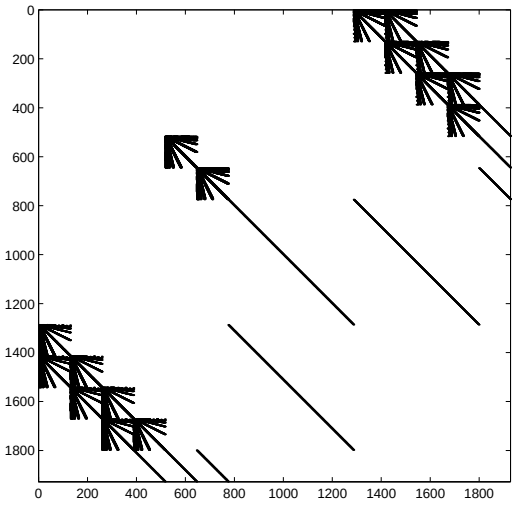
$$\text{subject to} \qquad \dot{\boldsymbol{x}} - \boldsymbol{A}\boldsymbol{x} - \boldsymbol{W}\boldsymbol{w} = \boldsymbol{B}\boldsymbol{u} \tag{18}$$

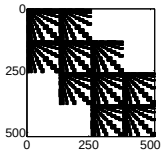
$$\boldsymbol{y} - \boldsymbol{C}\boldsymbol{x} - \boldsymbol{V}\boldsymbol{v} = \mathbf{0} \; . \tag{19}$$

However, the ideas can be extended to the nonlinear case with inequality constraints. Figure 4a 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 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

**Fig. 4.** a) KKT–matrix $\mathcal{L}_A$ b) Schur complement $\mathcal{S}_A$ .




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* [1,31]. Nevertheless, to exploit the problem structure as much as possible, we **reduce the system** and obtain a matrix $\tilde{\mathcal{L}}_A$, where we can apply the Uzawa algorithm (see [13]). This algorithm is particularly designed for saddle point problems with positive definite $\mathcal{A}_A$. Conceptionally it applies the *pcg*–method to the Schur complement and, to avoid inversion of $\mathcal{A}_A$, also to $\mathcal{A}_A$. In our case the inner application of the *pcg*–method is not necessary, since $\mathcal{A}_A^{-1}$ can actually be set up very efficiently. This means, that we apply essentially the *pcg*–method to the Schur complement $\mathcal{S}_A$ of the reduced system. For a detailed discussion of this **adapted Uzawa algorithm** we refer to [8]. In Figure 4b we can see the structure of the linear system after a reduction of $\mathcal{L}_A$ to the Schur complement $\mathcal{S}_A$. 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, $\mathcal{S}_A$ 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

$$\tilde{\boldsymbol{\mathcal{L}}}_{\Lambda^\ell} \begin{pmatrix} \tilde{\boldsymbol{\lambda}}_{\Lambda^\ell} \\ \tilde{\boldsymbol{\Upsilon}}_{\Lambda^\ell} \end{pmatrix} = \begin{pmatrix} \tilde{\boldsymbol{h}}_{\Lambda^\ell} \\ \tilde{\mathbf{g}}_{\Lambda^\ell} \end{pmatrix}, \quad \ell = \ell_0, \ell_0 + 1, \ldots, \tag{20}$$

with increasing dimension corresponding to the refined optimization problems.

The corresponding operator on the infinite dimensional function spaces

$$\tilde{\boldsymbol{\mathcal{L}}} : (L_2)^{n_y+n_w} \times (H^1)^{n_x} \longrightarrow (L_2)^{n_y+n_w} \times ((H^1)^{n_x})' \tag{21}$$

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 $\Lambda$. 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 $\boldsymbol{M}_{\Lambda^\ell}^{1/2}$ 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 $\Lambda^\ell$ independently of $\ell$ provided that the Galerkin scheme associated with these trial spaces is stable, i.e. the trial spaces satisfy the LBB–condition [13]. Consequently, we can deduce that the condition numbers fulfill

$$\kappa_\ell := \mathrm{cond}_2(\boldsymbol{M}_{\Lambda^\ell}^{1/2} \boldsymbol{\mathcal{S}}_{\Lambda^\ell} \boldsymbol{M}_{\Lambda^\ell}^{1/2}) \leq c \tag{22}$$

independently of $\ell$. The convergence rate of the adapted Uzawa algorithm can be estimated in terms of $\kappa_\ell$ 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 [8].

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 $\alpha$, (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 $\kappa_\ell$. These properties can be used to determine an upper bound for the number of iterations

$$max_{it} \approx log\,(2\bar{c}/\alpha) \,/\, \left(-log\,\left((\sqrt{\kappa}-1)/(\sqrt{\kappa}+1)\right)\right) \tag{23}$$

and a bound for the maximal relative residual error

$$\alpha_{rel} \approx \alpha/(\bar{c}\sqrt{\kappa}) \tag{24}$$

needed to produce on each level approximate solutions with discretization error accuracy (see [8]). 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 $\alpha_{rel}$ and $max_{it}$ an upper bound for $\kappa$ 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 [8] 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.

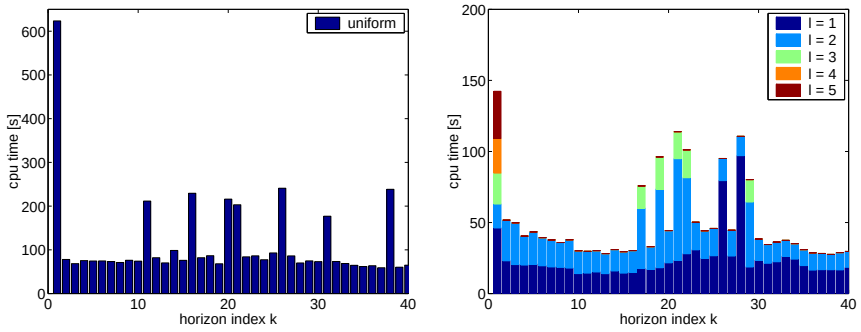
While the condition numbers provide an upper bound for the iteration numbers, the actual eigenvalue distributions give rise to considerably smaller iteration numbers. Superconvergence, known for the cg–method, can be observed for the adapted Uzawa algorithm, too. Nevertheless, more sophisticated methods 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 [8]. 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

**Fig. 5.** CPU times in each horizon. Uniform and adaptive discretization.




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 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. [33]). The process consists of an ideal continuous stirred tank reactor where a reversible exothermic reaction $A \rightleftharpoons R$ 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 [6]. The optimization problem is solved employing the active set based NLP solver SNOPT [26] 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 5. 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.

## References

1. Axelsson O.: *Iterative Solution Methods.* Cambridge University Press, New York, **1994**.
2. Binder T., Blank L., Bock H.G., Bulirsch R., Dahmen W., Diehl M., Kronseder T., Marquardt W., Schlöder J.P., v. Stryk O.: *Introduction to Model Based Optimization of Chemical Processes on Moving Horizons.* In Grötschel M., Krumke S.O., Rambau J. (editors): Online Optimization of Large Scale Systems: State of the Art, Springer–Verlag, Berlin, **2001**.

3. Binder T., Blank L., Dahmen W., Marquardt W.: *On the Regularization of Dynamic Data Reconciliation Problems.* RWTH Aachen, LPT–Report, LPT–2000-29, (submitted to J. of Proc. Contr.), **2000**.
4. Binder T., Blank L., Dahmen W., Marquardt W.: *Iterative Multiscale Methods for Process Monitoring.* RWTH Aachen, IGPM–Report 196, (to be published in: Proc. to "Fast Solution of Discretized Optimization Problems", WIAS, Berlin, Germany, 8-12.5.2000, ISNM, Birkäuser Verlag), **2000**.
5. Binder T., Blank L., Dahmen W., Marquardt W.: *Grid Refinement in Multiscale Dynamic Optimization.* In S. Pierucci (editor): Proc. European Symposium on Computer Aided Process Engineering, ESCAPE-10, Florence, Italy, 7-10.5.2000, 31-36, **2000**.
6. Binder T., Blank L., Dahmen W., Marquardt W.: *An Adaptive Multiscale Method for Real–Time Moving Horizon Optimization.* Proc. American Control Conference 2000, Chicago, USA, 28-30.6.2000, Omnipress, 4234-4238, **2000**.
7. Binder T., Blank L., Dahmen W., Marquardt W.: *Regularization of Dynamic Data Reconciliation Problems by Projection.* In Biegler L.T., Brambilla A., Scali C. (editors): Proc. IFAC International Symposium on Advanced Control of Chemical Processes, ADCHEM 2000, Pisa, Italy, 14-16.6.2000, Vol.2, 689-694, **2000**.
8. Binder T., Blank L., Dahmen W., Marquardt W.: *Iterative Algorithms for Multiscale Dynamic Data Reconciliation.* RWTH Aachen, IGPM–Report 186, (submitted to J. Opt. Th. Appl.), **2000**.
9. Binder T., Blank L., Dahmen W., Marquardt W.: *Adaptive Multiscale Approach to Real–Time Dynamic Optimization.* RWTH Aachen, LPT–Report, LPT–1999–14, **1999**.
10. Binder T., Blank L., Dahmen W., Marquardt W.: *Towards Multiscale Dynamic and Data Reconciliation.* In Berber R., Kravaris C. (editors): Nonlinear Model Based Process Control, NATO ASI, Kluwer Academic Publishers, 623-665, **1998**.
11. Binder T., Cruse A., Villar C., Marquardt W.: *Optimization using a Wavelet based Adaptive Control Vector Parametrization Strategy.* Comput. Chem. Eng. 24, 1201-1207, **2000**.
12. Bock H.G., Diehl M.M., Leineweber D.B., Schlöder J.P.: *A Direct Multiple Shooting Method for Real–Time Optimization of Nonlinear DAE Processes.* In Allgöwer F., Chen A. (editors): Nonlinear Model Predictive Control, Basel, Birkhäuser Verlag, 245-268, **2000**.
13. Braess D. *Finite Elemente.* Springer–Verlag, Berlin, **1992**.
14. Cervantes A., Biegler L.T.: *Large–scale DAE Optimization using a Simultaneous NLP Formulation.* AIChE J. 44(5), 1038-1050, **1998**.
15. Chambolle A., DeVore R., Lee N.Y., Lucier J.: *Nonlinear Wavelet Image Processing: Varational Problems, Compression, and Noise Removal through Wavelet Shrinkage.* IEEE Image Processing 7, 319-335, **1998**.
16. Chui C.K.: *An Introduction to Wavelets.* Academic Press, Boston, **1992**.
17. Cohen A., Daubechies I., Feauveau J.C: *Biorthogonal Bases of Compactly Supported Wavelets.* Comm. Pure and Appl. Math. 45, 485-560, **1992**.

18. Cuthrell J.E., Biegler L.T.: *On the Optimization of Differential–Algebraic Process Systems.* AIChE J. 33(8), 1257-1270, **1987**.

19. Dahmen W.: *Wavelet and Multiscale Methods for Operator Equations.* Acta Numerica, 55-228, **1997**.

20. Dahmen W., Schneider R., Xu Y.: *Nonlinear Functionals of Wavelet Expansions – Adaptive Reconstruction and Fast Evaluation.* Numer. Math. 86(1), 49-101, **2000**.

21. Daubechies I.: *Ten Lectures on Wavelets.* Society for Industrial and Applied Math., Philadelphia, **1992**.

22. DeVore R.A., Jawerth B., Popov V.A.: *Compression of Wavelet Decompositions.* Amer. J. Math. 114, 719-746, **1992**.

23. Donoho D., Johnstone I.: *Ideal Spatial Adaption by Wavelet Shrinkage*, Biometrika 81, 425-455, **1994**.

24. Donoho D., Johnstone I., Kerkyacharian G., Picard D.: *Wavelet Shrinkage: Asymptopia?* J. Roy. Statist. Assoc. 90, 301-369, **1995**.

25. Engl H.W., Hanke M., Neubauer A.: *Regularization of Inverse Problems* Kluwer, Dordrecht, The Netherlands, **1996**.

26. Gill P.E., Murray M., Saunders M.A.: *User's Guide for SNOPT 5.3: a Fortran Package for Large–Scale Nonlinear Programming.* Internal report, Department of Operations Research, Stanford University, **1997**.

27. Hansen C.: *Rank–Deficient and Discrete Ill–posed Problems: Numerical Aspects of Linear Inversion.* SIAM, Philadelphia, **1998**.

28. Hirschhorn R.M.: *Invertability of Multivariable Nonlinear Control Systems.* IEEE Trans. Auto. Cont., AC-24(6), 855-865, **1979**.

29. Kadam J.V.: *Index Analysis and Adaptive Refinement in Multiscale Dynamic Optimization.* Master Thesis, Lehrstuhl für Prozesstechnik, RWTH Aachen, **2000**.

30. Kirsch A.: *An Introduction to the Mathematical Theory of Inverse Problems.* Springer–Verlag, **1996**.

31. Meurant G.: *Computer Solution of Large Linear Systems.* Elsevier Science B.V., Amsterdam, The Netherlands, **1999**.

32. Neumaier A.: *Solving Ill–Conditioned and Singular Linear Systems: A Tutorial on Regularization.* SIAM Rev. 40, 636-666, **1998**.

33. Oliveira de S., Morari M.: *Contractive Model Predictive Control with Local Linearization for Nonlinear Systems.* In Berber R., Kravaris C. (editors): Nonlinear Model Based Process Control, NATO ASI, Kluwer Academic Publishers, 403-432, **1998**.

34. Plato R., Vainikko G.: *On the Regularization of Projection Methods for Solving Ill–posed Problems.* Numer. Math. 57, 63-79, **1990**.

35. Schittkowski K.: *Solving Constraint Nonlinear Least Squares Problems by a General Purpose SQP–method.* International Series of Numerical Mathematics 84, 295-309, **1988**.

36. Silverman L.M.: *Inversion of Multivariable Linear Systems.* Automatica, AC–14(3), 270-276, **1969**.

37. Tikhonov A.N., Goncharsky A.V., Stepanov V.V., Yagola A.G.: *Numerical Methods for the Solution of Ill–posed Problems.* Kluwer Academic Publishers, New York, **1996**.

38. Vassiliadis V.S., Sargent R.W.H., Pantelides C.C.: *Solution of a Class of Multistage Dynamic Optimization Problems. 1. Problems without Path Constraints.* Ind. Eng. Chem. Res. 33(9), 2111-2122, **1994**.

39. Walter C.: *Implementation and Evaluation of Interior–Point–Methods for the Solution of Wavelet Discretized Quadratical Optimal Control Problems.* Senior project, Lehrstuhl für Prozesstechnik, RWTH Aachen, **1999**.

40. Wright S.: *Primal–Dual Interior–Point Methods.* SIAM, **1997**.