Efficient dynamic optimization for Nonlinear Model Predictive Control - application to a High-Density Poly-Ethylene grade change problem

R.L. Tousain and O.H. Bosgra
Mechanical Engineering Systems and Control Group
Delft University of Technology, Mekelweg 2, 2628 CD, Delft
r.l.tousain@wbmt.tudelft.nl

Abstract

Nonlinear Model Predictive Control (NMPC) is believed to play an important role in improving the quality and flexibility of the production of many chemical plants. More widespread application can be expected when systematic solutions are found for modeling large-scale nonlinear processes and for efficient solution of the dynamic optimization problems NMPC entails. The control parametrization approach to dynamic optimization solves the dynamic optimization problem as a Nonlinear Program using e.g. the Sequential Quadratic Program (SQP) in the outer loop optimization problem. In the SQP approach, a reduced space Quadratic program is set up based on a quasi-Newton method estimate of the Hessian. We propose, based on an investigation of the structure of the Hessian of the NMPC problem, a different Hessian update procedure: part of the Hessian is calculated explicitly and only the part that relates to the second derivatives of the dynamics is estimated using a Hessian update. The proposed method shows a large improvement in computational efficiency for a semi-large-scale Poly-Ethylene reactor NMPC problem with 27 states and 6 inputs with 15 parameters each.

1 Introduction

Chemical industries are faced with rapidly changing market situations and high competition. The present-day market of chemicals is characterized by its enormous diversity and is highly demand-driven. To guarantee profitability of operation of existing production processes, often advances need to be made with respect to the quality and the flexibility of the production. One of the most popular and promising solutions in the field of process control is Model Predictive Control (MPC). Numerous successful implementations were reported (Froisy 1994), of which most are *linear* MPC applications. In respond to the industrial incentive to enable application of MPC on strongly nonlinear processes (e.g. batch or multi-product chemical plants such as HDPE plants), a lot of research on *Nonlinear* MPC (NMPC) has been done (see e.g. Henson (1998) for a review). However, widespread application of NMPC can only be expected when systematic solutions are found for modeling large-scale nonlinear processes and for efficient solution of the dynamic optimization problems that NMPC entails. This paper provides a contribution to the efficient solution of these optimization problems.

The characteristics of the dynamic optimization problems to be solved within NMPC have been studied extensively. Two basic solution methods for dynamic optimization can be distinguished: the *sequential* approach and the *simultaneous* approach. The sequential approach utilizes parametrization of the controls to discretize the problem (see e.g. Vassiliadis (1993)). An integration tool is used to evaluate the model equations and hence the objective function, the constraints and the gradients. Successive search directions towards the local optimum are determined by an outer loop optimizer, which is generally a Sequential Quadratic Programming tool (SQP). In the simultaneous approach, both the controls and the states are parametrized (Li et.al. 1989) to transform, mostly via collocation on finite elements, the dynamic optimization problem into a Nonlinear Program which can be solved using a Nonlinear Programming tool (e.g. SQP or the Generalized Reduced Gradient method (GRG)). The big advantage of the simultaneous approach is that the objective function and the process model equations converge simultaneously (infeasible path method), while the process model equations are necessarily satisfied in every iteration in the sequential approach. Also, the simultaneous approach can handle open loop unstable processes. However, the sequential approach is simpler in implementation and especially for stiff systems, it may actually be an advantage instead of a disadvantage that the model equations are satisfied every iteration. Another advantage of the sequential approach for on-line application is that intermediate solutions are feasible (with respect to the model equations). For these reasons, in our research we focus on the sequential approach.

The general purpose sequential approach to dynamic optimization can be relatively inefficient for problems with a specific and *known* structure such as the NMPC problem, which has only linear and quadratic terms in its objective functions. In this paper, we will show how the NMPC problem structure can be exploited by modification of the reduced space Hessian update of the SQP solver. The main idea is that instead of updating the full reduced space Hessian we can calculate part of the reduced space Hessian explicitly and estimate only the part that relates to the (unknown) nonlinear dynamics. A similar idea was posed in (Biegler 1992) for increasing the efficiency in simultaneous optimization methods for dynamic data reconciliation.

First, in Section 2 we introduce the NMPC problem formulation which is considered in this optimization study. Then, Section 3 explains how the standard SQP approach proceeds in solving the NMPC problem. In Section 4 the modified approach for dynamic optimization is motivated and described. Results on the optimization efficiency of both optimizers are shown for the case of a polymerization reactor grade change problem in Section 5. Finally, conclusions are given in Section 6.

2 Problem formulation

Without too much loss of generality, assume the dynamics of the process that is to be optimized to be described by a DAE system:

$$\dot{x} = f(x,u,v), \;\; 0 = g(x,u,v), \;\; y = C_y v, \qquad (1)$$

where $x \in \mathbb{R}^{nx}$, $v \in \mathbb{R}^{nv}$, and $u \in \mathbb{R}^{nu}$ are respectively the state, algebraic and input variables. $y \in \mathbb{R}^{ny}$ are the output variables, which appear in the objective or in the constraints. The general NLMPC objective J contains time-varying linear and quadratic weights on inputs and outputs and a linear-quadratic weight on the terminal state $x(t_f)$:

$$\begin{aligned} J \;=\; & F^T x(t_f) + x(t_f)^T Q x(t_f) + \textstyle\int_{t=0}^{t^f} \big(f_y(t)y(t) \\ & + y(t)^T q(t) y(t) + f_u(t)u(t) + u(t)^T r(t) u(t)\big)\, dt. \end{aligned} \qquad (2)$$

Input and output constraints are linear and possibly time-varying and defined as follows:

$$G_u(t)u < g_u(t), \;\; G_y(t)y < g_y(t), \qquad (3)$$

with $G_u \in \mathbb{R}^{nc_u \times nu}$, $g_u \in \mathbb{R}^{nc_u}$, $G_y \in \mathbb{R}^{nc_y \times ny}$ and $g_y \in \mathbb{R}^{nc_y}$ the constraint matrices, and with nc_u and nc_y the number of input respectively output constraints. Note that we do not model explicitly a terminal state constraint, which is in most NMPC formulations included for stability (see e.g. Henson (1998) for an overview). Extension with such a constraint is straightforward.

The receding horizon optimal control problem is defined as to minimize (2) subject to (1) and (3) over $u \in \mathcal{U}$, with $x(0) = \bar{x}_0$, and where $\mathcal{U} = L_\infty^{nu}[0,t_f) := \{f : [0,t_f) \to \mathbb{R}^{nu} \,|\, \sup_t |f_i(t)| < \infty, \forall i = 1,\ldots,nu\}$.

To discretize *optimal control problem I* we introduce the Riemann sum to approximate the integral objective. The discretized objective becomes:

$$\begin{aligned} J = F^T x_N + x_N^T Q x_N + \sum_{i=0}^{N-1} \tau \big(& f_{(y,i)} y_i + y_i^T q_i y_i \\ & + f_{(u,i)} u_i + u_i^T r_i u_i \big), \end{aligned} \qquad (4)$$

where N is the number of discretization intervals. The discretization is assumed time-uniform, with the length of each interval equal to $\tau = t_f/N$. Further, $u_i = u(i\tau)$, $y_i = y(i\tau)$, and $x_i = x(i\tau)$. We now assume that u is parametrized using zero-th order splines with length equal to τ. This is not a necessary assumption but it does simplify the description to follow. Then, if we introduce the operator $F_\tau(\cdot)$ for the integration of the DAE system over one time interval τ with constant input u_i the model equations can be written as

$$x_{i+1} = F_{t_s}(x_i, u_i), \;\; y_i = h(x_i, u_i) \qquad (5)$$

Further, we evaluate the constraints at times $0, \tau, 2\tau, 3\tau, \ldots, (N-1)\tau$:

$$G_{u,i} u_i < g_{u,i}, \;\; G_{y,i} y_i < g_{y,i}, \qquad (6)$$

for $i = 0, 1, 2, \ldots, N-1$. Then, the discretized optimal control problem can be defined as to minimize (4) subject to (5) and (6) over $p = [u_0^T \; u_1^T \; \ldots \; u_N]$, with $x_0 = \bar{x}_0$. This problem will be under consideration in the remainder of this section.

3 Standard sequential approach, using SQP

The conventional sequential solution approach is based on the discretization of the controls: $u = \theta(p,t)$ as in *control problem II*. An initial parameter vector is chosen. With the according input trajectories, the model is integrated using a numerical solver to obtain values for the objective function J, the constraints $\mathbf{g}$ and the gradients $\frac{\partial J}{\partial p}$ and $\frac{\partial \mathbf{g}}{\partial p}$. The optimizer calculates on the basis of the gradients a new search direction which is then implemented on the model. This scheme is executed iteratively until a local optimum is found.

3.1 Function evaluation

A shortcut notation for the objective evaluation is obtained by stacking the outputs and inputs in long vectors:

$$J = F^T x_N + x_N^T Q x_N + [\mathbf{f}_y^T \mathbf{y} + \mathbf{y}^T \mathbf{Q} \mathbf{y} + \mathbf{f}_u^T \mathbf{u} + \mathbf{u}^T \mathbf{R} \mathbf{u}], \qquad (7)$$

where

$$\begin{aligned}
\mathbf{y} &= [y_0^T\ y_1^T\ \dots\ y_{N-1}^T]^T,\\
\mathbf{u} &= [u_0^T\ u_1^T\ \dots\ u_{N-1}^T]^T,\\
\mathbf{f}_u &= \tau[f_{u,0}^T\ f_{u,1}^T\ \dots\ f_{u,N-1}^T]^T,\\
\mathbf{f}_y &= \tau[f_{y,0}^T\ f_{y,1}^T\ \dots\ f_{y,N-1}^T]^T,\\
\mathbf{Q} &= \tau\mathrm{diag}([q_0^T\ q_1^T\ \dots\ q_{N-1}^T]),\ \text{and}\\
\mathbf{R} &= \tau\mathrm{diag}([r_0^T\ r_1^T\ \dots\ r_{N-1}^T]).
\end{aligned}$$

Here, $a = \mathrm{diag}(b)$ means that a is a square matrix with the entries of vector b on the diagonal and zeros on the upper and lower diagonal triangular sub matrices.

A shortcut formulation of the constraints can also be achieved:

$$\mathbf{g} = \begin{bmatrix} \mathbf{G}_u & 0 \\ 0 & \mathbf{G}_y \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{y} \end{bmatrix} - \begin{bmatrix} \mathbf{g}_u \\ \mathbf{g}_y \end{bmatrix} < 0, \tag{8}$$

where $\mathbf{G}_u \in \mathbb{R}^{Nnc_u \times Nnu}$, $\mathbf{G}_y \in \mathbb{R}^{Nnc_y \times Nny}$, $\mathbf{g}_u \in \mathbb{R}^{Nnc_y}$, and $\mathbf{g}_y \in \mathbb{R}^{Nnc_y}$ are constructed through the combination of inequalities (6).

3.2 Gradient evaluation

In a general setting let the objective be given by the functional $J(\mathbf{y}(p), \mathbf{u}(p), x_N(p))$. Then, the objective gradient can be calculated as

$$\frac{\partial J}{\partial p} = \frac{\partial J}{\partial \mathbf{y}}\frac{\partial \mathbf{y}}{\partial p} + \frac{\partial J}{\partial \mathbf{u}}\frac{\partial \mathbf{u}}{\partial p} + \frac{\partial J}{\partial x_N}\frac{\partial x_N}{\partial p}. \tag{9}$$

For the NMPC objective, the Jacobian matrices $\frac{\partial J}{\partial \mathbf{y}} \in \mathbb{R}^{1\times Nny}$, $\frac{\partial J}{\partial \mathbf{u}} \in \mathbb{R}^{1\times Nnu}$, and $\frac{\partial J}{\partial x_N} \in \mathbb{R}^{1\times nx}$ can be calculated as follows.

$$\begin{aligned}
\tfrac{\partial J}{\partial \mathbf{y}} &= \mathbf{f}_y^T + 2\mathbf{y}^T\mathbf{Q},\\
\tfrac{\partial J}{\partial \mathbf{u}} &= \mathbf{f}_u^T + 2\mathbf{u}^T\mathbf{R},\\
\tfrac{\partial J}{\partial x_N} &= F^T + 2x_N^T Q.
\end{aligned} \tag{10}$$

The Jacobian matrix $\frac{\partial \mathbf{u}}{\partial p} \in \mathbb{R}^{Nny\times np}$ is calculated from the parametrization that is used. In our case, with the uniformly distributed zeroth order spline parametrization we have $\frac{\partial \mathbf{u}}{\partial p} = I$. $\frac{\partial \mathbf{y}}{\partial p} \in \mathbb{R}^{Nny\times np}$ and $\frac{\partial x_N}{\partial p} \in \mathbb{R}^{nx\times np}$ finally, the so-called "sensitivity matrices", represent a linearization of the DAE and can be calculated in several ways, which will be discussed later on in this section.

The constraint gradients can be calculated by differentiating (8) with respect to p which yields

$$\frac{\partial \mathbf{g}}{\partial p} = \begin{bmatrix} \mathbf{G}_u & 0 \\ 0 & \mathbf{G}_y \end{bmatrix} \begin{bmatrix} \frac{\partial \mathbf{u}}{\partial p} \\ \frac{\partial \mathbf{y}}{\partial p} \end{bmatrix}. \tag{11}$$

Calculation of the sensitivity matrix

One way to calculate $\frac{\partial \mathbf{y}}{\partial p}$ is to sample at the time instances $\{0, \tau, 2\tau, \dots, t_f\}$ the solution of the sensitivity equations (Støren et.al. 1995):

$$\begin{aligned}
\tfrac{\partial}{\partial t}\left(\tfrac{\partial x}{\partial p}\right)_t &= \tfrac{\partial f}{\partial x}\big|_t \left(\tfrac{\partial x}{\partial p}\right)_t + \tfrac{\partial f}{\partial u}\big|_t \left(\tfrac{\partial u}{\partial p}\right)_t + \tfrac{\partial f}{\partial v}\big|_t \left(\tfrac{\partial v}{\partial p}\right)_t,\\
0 &= \tfrac{\partial g}{\partial x}\big|_t \left(\tfrac{\partial x}{\partial p}\right)_t + \tfrac{\partial g}{\partial u}\big|_t \left(\tfrac{\partial u}{\partial p}\right)_t + \tfrac{\partial g}{\partial v}\big|_t \left(\tfrac{\partial v}{\partial p}\right)_t,\\
\left(\tfrac{\partial x}{\partial p}\right)_0 &= 0,
\end{aligned} \tag{12}$$

where $.|_t$ is a shortcut notation for $.|_{x=x(t),u=u(t),v=v(t)}$. The integration of the sensitivity equations can be very time-consuming. To avoid this, approximations to solving (12) can be used. For the NMPC optimization problem we propose an infinite order fixed step explicit integration scheme:

$$\begin{aligned}
\left(\tfrac{\partial x}{\partial p}\right)_{t+h} &= e^{A_t h}\left(\tfrac{\partial x}{\partial p}\right)_t + \textstyle\int_0^h e^{A_t s} ds B_t \left(\tfrac{\partial u}{\partial p}\right)_t,\\
\left(\tfrac{\partial \mathbf{y}}{\partial p}\right)_t &= C_t\left(\tfrac{\partial x}{\partial p}\right)_t + D_t\left(\tfrac{\partial u}{\partial p}\right)_t,
\end{aligned} \tag{13}$$

with

$$\begin{aligned}
A_t &= \tfrac{\partial f}{\partial x}\Big|_t - \tfrac{\partial f}{\partial v}\Big|_t \left(\tfrac{\partial g}{\partial v}\right)^{-1} \tfrac{\partial g}{\partial x}\Big|_t,\\
B_t &= \tfrac{\partial f}{\partial u}\Big|_t - \tfrac{\partial f}{\partial v}\Big|_t \left(\tfrac{\partial g}{\partial v}\Big|_t\right)^{-1} \tfrac{\partial g}{\partial u}\Big|_t,\\
C_t &= -C_y\left(\tfrac{\partial g}{\partial v}\Big|_t\right)^{-1} \tfrac{\partial g}{\partial x}\Big|_t,\\
D_t &= -C_y\left(\tfrac{\partial g}{\partial v}\Big|_t\right)^{-1} \tfrac{\partial g}{\partial u}\Big|_t,
\end{aligned} \tag{14}$$

with integration time step h. This explicit integration scheme is based on an assumption of local linearity of the dynamics in each interval of length h. The choice of h is a trade-off between integration accuracy and efficiency and is constrained from above by τ. For smoothly nonlinear processes, choosing h equal to τ may give satisfactory results, thence requiring only N evaluations of the Jacobian matrix; in other cases, we can choose h smaller or resort to implicit schemes at the expense of increased computation time. A practical advantage of this approach is that it enables to perform gradient evaluations external to the integration code and even on a different processor.

3.3 SQP optimization problem formulation

Applying the standard SQP optimization procedure (e.g. Nash et.al. (1996)) to the dynamic optimization problem on hand leads to the following iterative scheme (details are omitted):

step 1: set i, the iteration number to '0' and choose an initial parameter vector p^0, an initial Hessian matrix H^0 (normally chosen the identity matrix), and an initial λ^0; a proper estimate of H^0 and λ^0 can be taken from the previous NMPC execution to *warm start* the optimization in this cycle.
step 2: calculate the objective J^i and the constraints $\mathbf{g}^i$.
step 3: calculate the gradient of the reduced space Lagrangian $\nabla_p^i \mathcal{L} = \frac{\partial (J + \lambda^{iT}\mathbf{g})}{\partial p}(p)$ and of the constraints

$\nabla^i \mathbf{g} = \frac{\partial \mathbf{g}}{\partial p}(p)$ using respectively (9),(10), and (11).
step 4: if $i > 0$ update the reduced space Hessian estimate H^i based on $\nabla_p^i \mathcal{L}$ and $\nabla_p^{i-1} \mathcal{L}$ with a quasi-Newton approximation method.
step 5: solve the reduced space QP

$$\begin{array}{ll} \min\limits_{\Delta p} & [\nabla_p^i \mathcal{L}]^T \Delta p + \frac{1}{2} \Delta p^T H^i \Delta p \\ s.t. & [\nabla^i \mathbf{g}]^T \Delta p + \mathbf{g}^i \leq 0, \end{array} \tag{15}$$

for the search direction Δp, and update the estimate of the Lagrange multiplier.
step 6: Find a suitable search step $s^\star$ through the execution of a *line search* procedure such that $\mathcal{M}(p^i + s^\star \Delta p) < \mathcal{M}(p^i)$ for an appropriate Merit function $\mathcal{M}$. Set $p^{i+1} = p^i + s^\star \Delta p$, check the termination conditions and on failure proceed with **step 3**.

The *termination conditions* mentioned in **step 6** are generally expressed in terms of a maximum allowed change in the objective and the solution p in the last iteration.

4 Modified SQP approach

The standard implementation of SQP has a major drawback: the estimation of the Hessian is based on successive gradient evaluations only. As a consequence, if the estimated Hessian deviates significantly from the real Hessian, the quality of the search steps will be poor and many *costly* iterations will be required to converge to a local optimum. For these reasons we propose, in view of dynamic optimization for NMPC, a modification to the conventional SQP approach which does allow to use specific problem structure aiming to speed up convergence.

First, let us investigate the characteristics of the true reduced space Hessian of the NMPC problem:

$$H(p) = \begin{bmatrix} H_{1,1}(p) & H_{1,2}(p) & \cdots & H_{1,n_p}(p) \\ H_{2,1}(p) & H_{2,2}(p) & \cdots & \\ \vdots & \vdots & \ddots & \\ H_{n_p,1}(p) & & & H_{n_p,n_p}(p) \end{bmatrix}, \tag{16}$$

where $H_{k,l}(p) = \frac{\partial^2 J}{\partial p_k p_l}(p) + \lambda^T \frac{\partial^2 \mathbf{g}}{\partial p_k p_l}(p)$. Using $\frac{\partial J}{\partial p_k} = \sum_{i=1}^{Nny} \frac{\partial J}{\partial \mathbf{y}_i} \frac{\partial \mathbf{y}_i}{\partial p_k} + \sum_{i=1}^{Nnu} \frac{\partial J}{\partial \mathbf{u}_i} \frac{\partial \mathbf{u}_i}{\partial p_k} + \sum_{i=1}^{nx} \frac{\partial J}{\partial x_N(i)} \frac{\partial x_N(i)}{\partial p_k}$ (compare (9)) we can derive

$$\begin{aligned} \frac{\partial^2 J}{\partial p_k p_l} &= \sum_{i=1}^{Nny} \left(\sum_{j=1}^{Nny} \frac{\partial^2 J}{\partial \mathbf{y}_i \partial \mathbf{y}_j} \frac{\partial \mathbf{y}_j}{\partial p_l} \frac{\partial \mathbf{y}_i}{\partial p_k} \right) + \frac{\partial J}{\partial \mathbf{y}_i} \frac{\partial^2 \mathbf{y}_i}{\partial p_k \partial p_l} \\ &+ \sum_{i=1}^{Nnu} \left(\sum_{j=1}^{Nnu} \frac{\partial^2 J}{\partial \mathbf{u}_i \partial \mathbf{u}_j} \frac{\partial \mathbf{u}_j}{\partial p_l} \frac{\partial \mathbf{u}_i}{\partial p_k} \right) + \frac{\partial J}{\partial \mathbf{u}_i} \frac{\partial^2 \mathbf{u}_i}{\partial p_k \partial p_l} \\ &+ \sum_{i=1}^{nx} \left(\sum_{j=1}^{nx} \frac{\partial^2 J}{\partial x_N(i) \partial x_N(j)} \frac{\partial x_N(j)}{\partial p_l} \frac{\partial x_N(i)}{\partial p_k} \right) \\ &+ \frac{\partial J}{\partial x_N(i)} \frac{\partial^2 x_N(i)}{\partial p_k \partial p_l}. \end{aligned} \tag{17}$$

Further, using $\frac{\partial \mathbf{g}_m}{\partial p_k} = \sum_{i=1}^{Nny} \frac{\partial \mathbf{g}_m}{\partial \mathbf{y}_i} \frac{\partial \mathbf{y}_i}{\partial p_k} + \sum_{i=1}^{Nnu} \frac{\partial \mathbf{g}_m}{\partial \mathbf{u}_i} \frac{\partial \mathbf{u}_i}{\partial p_k}$ we find, for constraints which are linear in y and u:

$$\frac{\partial^2 \mathbf{g}_m}{\partial p_k \partial p_l} = \sum_{i=1}^{Nny} \frac{\partial \mathbf{g}_m}{\partial \mathbf{y}_i} \frac{\partial^2 \mathbf{y}_i}{\partial p_k \partial p_l} + \sum_{i=1}^{Nnu} \frac{\partial \mathbf{g}_m}{\partial \mathbf{u}_i} \frac{\partial \mathbf{u}_i^2}{\partial p_k \partial p_l}. \tag{18}$$

Based on this understanding of the structure of the reduced space Hessian, we now propose to partition the Hessian as follows:

$$H(p) = H^c(p) + H^d(p), \tag{19}$$

where H^c contains the second derivatives of the objective and only first derivatives of the dynamics:

$$\begin{aligned} H^c_{(k,l)}(p) &= \sum_{i=1}^{Nny} \left(\sum_{j=1}^{Nny} \frac{\partial^2 J}{\partial \mathbf{y}_i \partial \mathbf{y}_j} \frac{\partial \mathbf{y}_j}{\partial p_l} \frac{\partial \mathbf{y}_i}{\partial p_k} \right) \\ &+ \sum_{i=1}^{Nnu} \left(\sum_{j=1}^{Nnu} \frac{\partial^2 J}{\partial \mathbf{u}_i \partial \mathbf{u}_j} \frac{\partial \mathbf{u}_j}{\partial p_l} \frac{\partial \mathbf{u}_i}{\partial p_k} \right) \\ &+ \sum_{i=1}^{nx} \left(\sum_{j=1}^{nx} \frac{\partial^2 J}{\partial x_N(i) \partial x_N(j)} \frac{\partial x_N(j)}{\partial p_l} \frac{\partial x_N(i)}{\partial p_k} \right), \end{aligned} \tag{20}$$

and H^d contains the second derivatives of the dynamics and only first derivatives of the objective:

$$\begin{aligned} H^d_{(k,l)}(p) &= \sum_{i=1}^{Nny} \frac{\partial J}{\partial \mathbf{y}_i} \frac{\partial^2 \mathbf{y}_i}{\partial p_k \partial p_l} + \sum_{i=1}^{Nnu} \frac{\partial J}{\partial \mathbf{u}_i} \frac{\partial^2 \mathbf{u}_i}{\partial p_k \partial p_l} \\ &+ \sum_{i=1}^{nx} \frac{\partial J}{\partial x_N(i)} \frac{\partial^2 x_N(i)}{\partial p_k \partial p_l} \\ &+ \sum_{m=1}^{N(nc_u + nc_y)} \lambda_m \left(\sum_{i=1}^{Nny} \frac{\partial \mathbf{g}_m}{\partial \mathbf{y}_i} \frac{\partial^2 \mathbf{y}_i}{\partial p_k \partial p_l} + \right. \\ &\qquad \left. \sum_{i=1}^{Nnu} \frac{\partial \mathbf{g}_m}{\partial \mathbf{u}_i} \frac{\partial^2 \mathbf{u}_i}{\partial p_k \partial p_l} \right). \end{aligned} \tag{21}$$

$H^c(p)$ can be calculated explicitly based on the gradient of the dynamics in combination with the analytic expression of the cost functional. For the NMPC problem discussed in this chapter it can be easily verified that

$$H^c(p) = \frac{\partial x_N}{\partial p}^T Q \frac{\partial x_N}{\partial p} + \frac{\partial \mathbf{y}}{\partial p}^T \mathbf{Q} \frac{\partial \mathbf{y}}{\partial p} + \mathbf{R}. \tag{22}$$

For systems with weakly nonlinear dynamics, using $H(p) = H^c(p)$ as an approximation of the reduced space Hessian in every iteration may yield adequate search directions (this approach shows strong similarities to the Gauss-Newton approach for solving nonlinear least-squares problems). For systems that do not satisfy this "weak-nonlinearity condition" performance may be poor using this approximation of the reduced space Hessian which motivates the inclusion (in some way) of the contribution of the second order derivatives of the dynamics in the approximation. We propose to construct (and update) this contribution based on successive evaluations of the gradient like in the conventional SQP approach discussed before. Applying the BFGS update yields:

$$H^{d,i+1} = H^{d,i+1} - \frac{(H^{d,i} s^i)(H^{d,i} s^i)^T}{s^{iT} H^{d,i} s^i} + \frac{\gamma^i \gamma^{iT}}{\gamma^{iT} s^i}, \tag{23}$$

where $s^i = p^{i+1} - p^i$, and $\gamma^i = \nabla_p^{i+1} \mathcal{L} - \nabla_p^i \mathcal{L} - H^{c,i+1} s^i$. Other update schemes can be used when deemed appropriate.

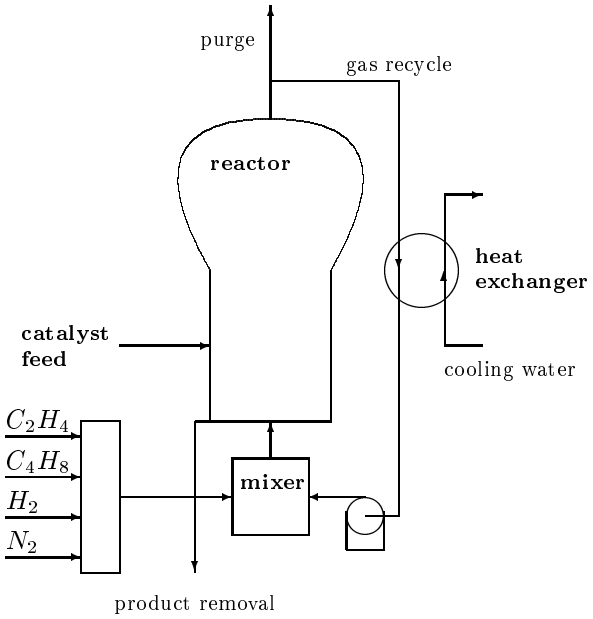

Figure 1: *Schematic process flow sheet of the gas phase HDPE polymerization process*

5 Example

We consider the implementation of NMPC for grade change control of a High-Density Poly-Ethylene reactor. A similar process was studied by (McAuley et.al. 1992) in their optimization studies. A schematic flow sheet of the polymerization reactor considered is shown in Figure 5. The gaseous monomers, Ethylene and Butene, together with Hydrogen and Nitrogen enter the reactor at the bottom. Only part of the gas flow reacts on the catalyst surface in the fluidized bed; the rest leaves the reactor through the gas cap and is recycled through the counter current flow heat exchanger in which the heat of reaction is removed. Part of the content of the reactor can be flared through the purge. The reactions are modeled according to Ziegler-Natta kinetics. The NMPC is used as a supervisory controller on top of a basic control systems consisting of a temperature controller, a pressure controller, a level controller and a ratio controller for the feeds. The model available for optimization studies is similar to the one used in (Schot et al. 1999). For this study, we approximated this model by a Wiener nonlinear model with 27^{th} order linear dynamics and static nonlinearities in the outputs. The inputs and outputs considered are, together with their respective bounds given in Table 1.

The grade change optimization problem is defined as follows: to switch the plant from producing a certain polymer grade A (density: 933.6 kg/m^3, log(melt index): -0.44) to producing a polymer grade B (density: 930.2 kg/m^3, log(melt index): 7.15) in 7 hours with a user-defined compromise between low costs and low control energy. The transition time is assumed to be determined from an off-line dynamic optimization with

mv/cv	description	lb	ub
$u(1)$	extra but. feed $[kg/hr]$	-	-
$u(2)$	extra hydr. feed $[kg/hr]$	-	-
$u(3)$	nitr. feed $[kg/hr]$	0	
$u(4)$	cat. feed $[kg/hr]$	0	1
$u(5)$	bed level setpoint $[m]$	5	7
$u(6)$	purge flow $[kg/hr]$	0	10
$y(1)$	density $[kg/m^3]$ $t \in [0,4]$	932.6	934.6
	$t \in [11,15]$	929.2	931.2
$y(2)$	log(melt index) $t \in [0,4]$	-1.4	0.6
	$t \in [11,15]$	6.15	8.15
$y(3)$	production $[kg/hr]$	2000	5000
$y(4)$	cooling water flow $[kg/hr]$	1000	52000
$y(5)$	eth. feed $[kg/hr]$	0	5000
$y(6)$	but. feed $[kg/hr]$	0	1000
$y(7)$	hydr. feed $[kg/hr]$	0	100
$y(8)$	nitr. mass fr. gas phase $[-]$	0.6	0.7

Table 1: manipulated variables and controlled variables with their respective upper and lower bounds.

a more complex and stronger non-linear objective (how such a problem can be formulated and solved is described in Schot (1999)). The NLMPC objective is a mixed economic-control objective: $J = scJ^{ec} + J^{co}$, with weighting factor sc. The economic objective is defined as follows:

$$\begin{aligned} J^{ec} &= \textstyle\sum_{i=0}^{14} p_{eth} y_5 + p_{but} y_6 + p_{hyd} y_7 + p_{nit} y_8 \\ &- \textstyle\sum_{i=0}^{3} p_{polA} y_3 - \sum_{i=4}^{10} p_{poloff} y_3 - \sum_{i=11}^{14} p_{polB} y_3, \end{aligned} \tag{24}$$

where p_{polA} is the price of polymer grade A and p_{polB} is the price of polymer grade B. p_{poloff} is the price of off-spec polymer which is in this case chosen lower than the price of the raw material; hence, there is a strong incentive to decrease the production during the transition. The control objective is chosen such that product quality variance and control energy are penalized:

$$\begin{aligned} J^{c} &= \textstyle\sum_{i=0}^{1} 4 \sum_{j=1}^{6} u_i(j)^T r_j u_i(j) \\ &+ \textstyle\sum_{i=0}^{3} \sum_{j=1}^{8} q_j (y_i(j) - y_{set}^A(j))^2 \\ &+ \textstyle\sum_{i=11}^{14} \sum_{j=1}^{8} q_j (y_i(j) - y_{set}^B(j))^2, \end{aligned} \tag{25}$$

with $r = [1.10^{-5}\ \ 1.10^{-3}\ \ 1.10^{-4}\ \ 1.10^{-3}\ \ 1\ \ 1.10^{-5}]$, $q = [1\ 1\ 0\ 0\ 0\ 0\ 0\ 0]$, and $y_{set}^A(1) = 933.6$, $y_{set}^A(2) = -0.44$, $y_{set}^B(1) = 930.2$, $y_{set}^B(2) = 7.15$. No end-weighting was defined since the horizon length is long enough to reach steady-state.

The mv's are parametrized as piece-wise constant inputs with 15 parameters each. The optimization is done using the standard SQP approach first. For $sc = 0.5$ more than 50 iterations were required before the solution tolerance was within the termination criterion. The solution found was -6.5848. Convergence was especially slow close to the solution. The modified SQP approach required only 17 iterations to converge to the same optimum. The solver progress is plotted in Table 2 for both methods for the first 20 iterations of both optimization cycles. As can be seen, convergence of the modified SQP approach is much faster. The trajectories of the most significant cv's are shown in Figure 2 for solutions of the optimization problem with three

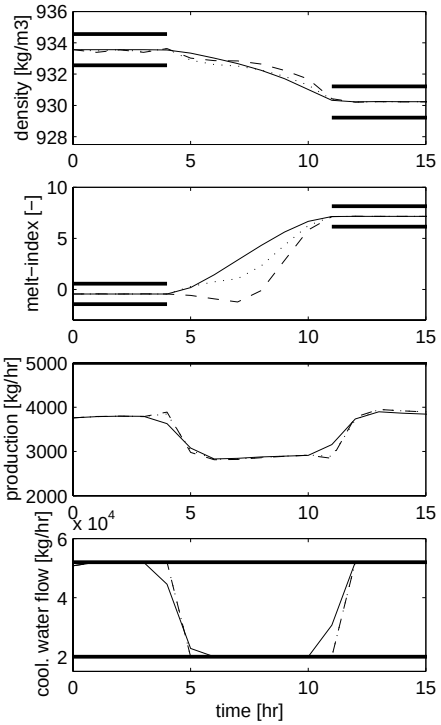

Figure 2: Result of the NMPC optimization grade transition control for a HDPE reactor. solid line: $sc = 0.01$, dotted line: $sc = 0.5$, dashed line: $sc = 1$.

values of sc. Constraints are indicated with bold lines. As expected, a stronger weighting of the economic objective leads to more aggressive changeovers, where the decrease in (money-wasting) off-spec production during the transition is rudely enforced.

6 Conclusions

In this paper, we discussed the characteristics of the dynamic optimization problem one encounters when applying Nonlinear Model Predictive Control. The general purpose Sequential Quadratic Programming approach to solving such problems appears to be inefficient for this class of problems, since the special structure of the Hessian is not exploited. The SQP can be tailored to NMPC applications by modifying the SQP Hessian update algorithm: the part that relates to the second order terms in the objective is calculated explicitly and only the second order dynamics terms are estimated using a secant update method. The proposed method shows a large improvement in computational efficiency for a semi-large-scale Poly-Ethylene reactor NMPC problem with 27 states and 6 inputs with 15 parameters each.

iteration	function value	
	conventional SQP	modified SQP
1	63.2788	63.2788
2	-1.1759	-3.0606
3	-3.3057	-5.4088
4	-5.1138	-6.2683
5	-5.3091	-6.5682
6	-5.5545	-6.5706
7	-6.0191	-6.5739
8	-6.4060	-6.5814
9	-6.4394	-6.5825
10	-6.5151	-6.5831
11	-6.5256	-6.5837
12	-6.5295	-6.5840
13	-6.5310	-6.5842
14	-6.5347	-6.5844
15	-6.5386	-6.5846
16	-6.5395	-6.5847
17	-6.5401	-6.5848
18	-6.5407	-6.5848
19	-6.5419	[-]
20	-6.5430	[-]

Table 2: Solver progress for the first 20 iterations for the mixed economic-control objective NMPC problem with $sc = 0.5$.

References

Biegler, L.T. (1992). Tailoring optimization algorithms to process applications. *Computers chem. Engng.*, Vol. 16, suppl., 81-95.

Froisy, B.J. (1994), Model predictive control: Past, present and future, *ISA Transactions*, Vol. 33, 235-243.

Henson, M.A. (1998). Nonlinear model predictive control: current status and future directions, *Computers and Chemical Engineering*, Vol. 23, 187-202.

Li, W.C. and L.T. Biegler (1989). Multistep, Newton-type control strategies for constrained, nonlinear processes, *Chem. Engng. Res. Dev.*, Vol. 67, 562-577.

McAuley, K.B. and J.F. MacGregor (1992), Optimal grade transitions in a gas phase polyethylene reactor. *Aiche Journal*, Vol. 38, n. 10, 1564-1576.

Nash, S.G. and A. Sofer (1996). *Linear and nonlinear programming*, McGraw-Hill Book Company.

Schot, J.J. van der, Tousain, R.L., Backx, A.C.P.M., and Bosgra, O.H. (1999). SSQP for the solution of large-scale dynamic-economic optimization problems. *Computers and Chemical Engineering Supplement*, S507-S510.

Støren, S. and T. Hertzberg (1994), The sequential linear quadratic programming algorithm for solving dynamic optimization problems - a review, *Computers chem. Engng.*, Vol. 19, 495-500.

Vassiliadis, V. (1993). *Computational solution of dynamic optimization problems with general differential-algebraic constraints*, Phd. thesis, Imperial College of Science.