Observer Based Restricted Structure Generalized Predictive Control for quasi-LPV Nonlinear Systems

Mike J. Grimble*** and Pawel Majecki**

* University of Strathclyde, Glasgow (e-mail:m.j.grimble@strath.ac.uk)
** ISC Limited, Culzean House, 36 Renfield Street, Glasgow G2 1LU, UK (e-mail: pawel@isc-ltd.com)

Abstract: An observer based Restricted Structure Generalized Predictive Control (RS-GPC) algorithm is proposed. The novel feature is to assume the state-observer within the feedback loop is of reduced order. The aim is to inherit the natural robustness of low-order controllers and to provide a solution that may be easily simplified for real-time implementation. The nonlinear discrete-time, multivariable plant model is represented by a state-space system that may be in Linear Parameter Varying or State-Dependent forms. The controller gains are computed to minimize the type of cost-function that is found in traditional model predictive control but with some additional terms that enable gain magnitudes and the rate of change of control gains to be minimized. The cost-function also includes dynamically weighted tracking-error and control signal costing terms. The optimal controller includes a reduced order observer and a time-varying control gain matrix within the loop and background processing for the gain computations. Hard constraints may be imposed on the gain and rate of change of gain and on the control and output signals.

Keywords: Observer, predictive, restricted structure, optimal, Linear Parameter Varying, state-dependent

1 INTRODUCTION

There have been many attempts to derive “low-order” restricted structure optimal controllers using direct optimization of parameters, such as the early work of Vladimir Zakian (1973) and Lucien Polak (1982) and colleagues, and the more recent work of Hast (2015) and Grimble (2004). The Quantitative Feedback Theory (QFT) design method provides a low-order robust control solution (Houpis et al., 2006). Richalet (1993) also developed an approach to predictive control that attempts to simplify the solution. For applications, Sato (2010) used a classical PID controller structure with a predictive approach for weigh feeder control. Eielsen et al., (2013), compared five fixed structure low-order controllers for Nano positioning systems. However, most advanced controllers used in applications have a Kalman filtering separation structure; optimal or suboptimal.

Observer based control structures like the linear quadratic Gaussian problem solution where an optimal observer or Kalman filter is involved are very popular. The structure is also used in many forms of nonlinear control law, even when the "separation principle" does not apply. There are many benefits of an observer-based solution, such as the ability to run the estimator without closing the control loop during commissioning. However, there is a disadvantage with an optimal control approach. That is, the order of the controller to be implemented within the feedback loop is often high. It is equal to the number of states in the combined plant model, disturbances, reference model and any cost-function dynamic weightings.

The approach proposed here is to use a reduced-order observer or filter within the feedback loop that provides the state estimates \( \hat{x}(t) \). This observer can be based on a reduced-order model of the plant, obtained by system identification or by using model order reduction. The state-estimate feedback gain is found to minimise a predictive control cost-index on-line. The low-order observer or filter is assumed to have a form that is pre-specified. The optimal low-order “observer based” feedback controller that is obtained and implemented within the feedback loop is therefore of a Restricted Structure (RS). It relegates the optimization computations to background processing.

Model Predictive Control (MPC) and so-called Generalized Predictive Control (GPC), is very successful in industry in its various model-based algorithm forms. The particular form of MPC algorithm is referred to as Restricted Structure Generalized Predictive Control (RS-GPC). The control is parameterized to have a state-estimate solution \( u(t) = k_c(t)\hat{x}_c(t) \) where \( \hat{x}_c(t) \) denotes the estimate of states from the low-order observer. It involves the online calculation of the gains \( k_c(t) \), rather than the computation of the vector of future controls in “implicit MPC.” Hard and soft constraints may be applied to the state-estimate feedback-control gain terms, and to the rate of change of gains. There are options to simplify real-time implementation, such as storing the controller gains at different operating points and then using a scheduling algorithm as in “explicit MPC.”

Roadmap: The \( qLPV \) state-space model for the system is described in § 2 and the RS controller structure is introduced in § 3. The cost-function and the solution of the RS-GPC problem is considered in § 4 and an application is presented in § 5.

2 SYSTEM DESCRIPTION

The discrete-time multivariable plant model is represented by a quasi Linear Parameter Varying (\( qLPV \)), state-space model that can be open loop unstable. The system in Fig. 1 includes the plant, reference, noise and disturbance signals. The \( qLPV \) state-space model of the plant is assumed to be pointwise...
stabilizable and detectable and can be represented by an operator:

\[(Y_{u0}u_0)(t) = z^{-1} (V_{u0}u_0)(t) \]

(1)

where the term \( z^{-1} I \) denotes a diagonal matrix of delay elements with \( k > 0 \). The disturbance signal is a qLPV model driven by zero-mean white noise and a deterministic output disturbance component denoted \( d(t) \). The reference \( r(t) \) is deterministic, and the deterministic signals are known in the prediction horizon. The white measurement noise \( v(t) \) has a constant covariance matrix \( R_y = R^2 \geq 0 \) and the zero-mean white noise source \( \xi(t) \) has an identity covariance matrix.

Measurement noise \( v(t) \) and driving noise \( \xi(t) \) are also assumed zero-mean, independent, white noise.

\[ x(t + i) = A_i x(t) + B_i u_0(t - k) + D_i \xi(t) + d_i(t) \]

(2)

The known disturbance term \( d_i(t) \) is \( 0 \) for \( i = 0 \):

\[ d_i(t) = \sum_{j=1}^{i} \sum_{j=1}^{i} A_{i,j} A_{i-1,j} \ldots A_{i-j} \xi(t) \]

(6)

and if \( j = i \) term \( A_{i,i} A_{i+1,i-1} \ldots A_{i,j} \) is defined as the identity.

The future states may be obtained from (6) as:

\[ x(t + i + k) = d_{i+k}(t + i + k - 1) + A_{i+k} \ldots A_{i} x(t + k) \]

(8)

Collecting the deterministic disturbance signal terms together:

\[ d_{i+k}(t + i + k) = d_{j}(t + i + k) + C_{p_{i+k}} d_{i+k}(t + i + k) \]

(9)

Introduce the notation: \( A_{i,m} = A_{i+1,i-1} \ldots A_{i} \) for \( i > m \), \( A_{i,m} = 1 \) for \( i = m \) and \( A_{i,m} = A_{i+1,i-1} \ldots A_{i} \) for \( i > 0 \), \( A_{i,m} = 1 \) for \( i = 0 \), where \( A_{i,j}^{m} = A_{i+1,j} A_{i+2,j} \ldots A_{i,j}^{m} \). Using this notation, the future tracking error:

\[ e_{i}(t + i + k) = d_{i+k}(t + i + k) + E_{i+k} u_0(t + i) + C_{i+k} A_{i} x(t + k) \]

(10)

2.3 Vector Matrix Prediction Equation

Introducing an obvious notation these signals they may be collected in the \( N+1 \) vector form (Ords and Clarke, 1993) as:

\[ E_{P_{i+k},N} = \sum_{j=1}^{i} \sum_{j=1}^{i} C_{P_{i+k},j} A_{i,j} \xi(t + k) \]

\[ + \sum_{j=1}^{i} \sum_{j=1}^{i} C_{P_{i+k},j} A_{i,j} \xi(t + j + k - 1) \]

(12)

For \( N \geq 1 \), the \( N+1 \) square block matrices follow as:

\[ E_{P_{i+k},N} = \text{diag} \{ C_{P_{i+k}}, \ldots, C_{P_{i+k}} \} \]
\[ E_{P_{i+k,N}} = \text{diag}\{E_{P_{i+k+1}}, E_{P_{i+k+2}}, \ldots, E_{P_{i+N+1}}\} \] (N+I square)

\[
A_{i+k,N} = \begin{bmatrix}
I & 0 & 0 & \cdots & 0 \\
A_{i+k} & \mathcal{B}_{i+k} & 0 & \cdots & 0 \\
\vdots & \vdots & \ddots & \ddots & \vdots \\
A_{i+k} & \mathcal{B}_{i+k} & 0 & \cdots & 0 \\
\end{bmatrix}, \
B_{i+k,N} = \begin{bmatrix}
0 & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 0 \\
\end{bmatrix}
\]

\[
D_{i+k,N} = \begin{bmatrix}
D_{i+k} & 0 & \cdots & 0 \\
D_{i+k+1} & 0 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
D_{i+k+1} & 0 & \cdots & 0 \\
\end{bmatrix}, \
\xi(t), E_{P_{i+k,N}} = \begin{bmatrix}
e_p(t) \\
e_p(t+1) \\
\vdots \\
e_p(t+N) \\
\end{bmatrix}
\]

\[
\xi(t), d_p(t) = \begin{bmatrix}
u_0(t) \\
u_0(t+1) \\
\vdots \\
u_0(t+N) \\
\end{bmatrix}, \quad U_{i,N} = \begin{bmatrix}
u_0(t) \\
u_0(t+1) \\
\vdots \\
u_0(t+N) \\
\end{bmatrix}, \quad D_{i+k,N} = \begin{bmatrix}d_p(t) \\
d_p(t+1) \\
\vdots \\
d_p(t+N) \\
\end{bmatrix}
\]

\[
\Xi_{i,N} = \begin{bmatrix}
[\mathcal{A}_{i+k,N} \mathcal{B}_{i+k,N} \ldots \mathcal{B}_{i+k,N-1}, 0]
\end{bmatrix}
\]

Matrix \( \Xi_{i,N} \) denotes a vector of future white noise inputs, \( U_{i,N} \) denotes a block-vector of predicted controls and \( U_{i,N} - U_{i,N} \) is a vector of future reference controls. From (8) the end-state:

\[
x(t + N + k) = d_{i+k}(t + N + k) + A_{i+k,N} x(t + k) + B_{i+k,N} u_0(t + k)
\]

where \( \mathcal{B}_{i+k,N} = \begin{bmatrix}A_{i+k+1,N} \mathcal{B}_{i+k} \ldots \mathcal{B}_{i+k,N-1}, 0 \end{bmatrix} \). The end-state error:

\[
e_p(t) = e_p(t) + \mathcal{B}_i u_0(t + k) + d_{i+k}(t)
\]

\[
\begin{align*}
- & d_{i+k}(t + N + k) - A_{i+k,N} x(t + k) - \mathcal{B}_{i+k,N} u_0(t + k) - \Xi_{i,N} z(t + k) \\
& = e_p(t) + \mathcal{B}_i u_0(t + k) + d_{i+k}(t)
\end{align*}
\]

The \( k \) steps-ahead tracking error \( E_{P_{i+k,N}} \) includes any dynamic weighting and may be written, using (12), as:

\[
E_{P_{i+k,N}} = D_{i+k,N} + C_{P_{i+k,N}} A_{i+k,N} x(t + k) + C_{P_{i+k,N}} B_{i+k,N} u_0(t + k) + \Xi_{i,N} z(t + k)
\]

\[
= \mathcal{D}_{i+k,N} + C_{P_{i+k,N}} A_{i+k,N} x(t + k) + \mathcal{U}_{i+k,N} u_0(t + k) + \Xi_{i,N} z(t + k)
\]

\[
\mathcal{Y}_{P_{i+k,N}} = \mathcal{C}_{P_{i+k,N}} \mathcal{B}_{i+k,N} + \mathcal{E}_{P_{i+k,N}}
\]

\[ 2.4 \text{ Prediction Equations} \]

The \( i \)-steps ahead prediction of the weighted error signal may be found by noting (11), assuming future controls are known. Let the prediction error \( \hat{e}_p(t+i+k \mid t) = E[\hat{e}_p(t+i+k) \mid t] \) then:

\[
\hat{e}_p(t+i+k \mid t) + C_{P_{i+k,N}} A_{i+k,N}^{i} \hat{e}(t+j+k \mid t) + \sum_{j=0}^{i} C_{P_{i+k,N}} A_{i+k,N}^{i-j} B_{i+k} \mathcal{u}_0(t+j) + \mathcal{Z}_{i+k,N} \]

\[ \hat{e}_p(t+i+k \mid t) = d_{i+k}(t+i+k) + E_{P_{i+k,N}} u_0(t+i) + \mathcal{Z}_{i+k,N} \]

Noting (16), the vector of prediction errors \( \hat{E}_{P_{i+k,N}} \) follows as:

\[
\hat{E}_{P_{i+k,N}} = \mathcal{D}_{i+k,N} + C_{P_{i+k,N}} A_{i+k,N} \hat{e}(t+k \mid t) + \mathcal{U}_{i+k,N} \mathcal{u}_0(t+k - N)
\]

The estimation error \( \hat{E}_{P_{i+k,N}} = E_{P_{i+k,N}} - \hat{E}_{P_{i+k,N}} \) follows as:

\[
\hat{E}_{P_{i+k,N}} = C_{P_{i+k,N}} \mathcal{A}_{i+k,N} x(t+k) + C_{P_{i+k,N}} A_{i+k,N} \hat{e}(t+k \mid t) = C_{P_{i+k,N}} \mathcal{A}_{i+k,N} x(t+k \mid t) + \mathcal{E}_{P_{i+k,N}} \mathcal{u}_0(t+k - N) + \mathcal{Z}_{i+k,N}
\]

where state estimation error \( \hat{e}(t+k \mid t) = x(t+k) - \hat{x}(t+k \mid t) \).

If the plant and model mismatch is ignored the state estimation error is independent of the choice of control. Recall \( \hat{x}(t+k \mid t) \) and \( \hat{x}(t+k \mid t) \) are orthogonal and the expectation of the product of the future values of the control action (assumed known in the prediction equation), and the white noise driving signals, is null. It follows \( \hat{E}_{P_{i+k,N}} \) in (19) and \( \hat{E}_{P_{i+k,N}} \) are orthogonal.

\[ \text{Time-Varying Kalman Estimator:} \]

The Kalman filter algorithm may be written in Predictor and Corrector form as:

\[
\hat{x}(t+1 \mid t) = A_{i+k} \hat{x}(t \mid t) + B_{i+k} u_0(t-k) + d_{i+k}(t)
\]

\[
\hat{x}(t+1 \mid t+1) = \hat{x}(t+1 \mid t) + C_{i+k+1} \hat{x}(t+1) \]

\[
\hat{z}(t+1 \mid t) = d(t+1) + C_{i+k+1} \hat{x}(t+1) + E_{i+k+1} u_0(t+1 - k)
\]

Observe that \( \hat{x}(t+k \mid t) \) only depends upon past values of the control signal. This is a full-order optimal estimator with different states to those of the reduced-order observer \( \hat{x}_p(t \mid t) \).

\[ 3 \text{ OBSERVER BASED RS PREDICTIVE CONTROL} \]

The GPC cost-index that motivates the criterion to be minimised, can contain a dynamic error weighting (Ordys and Clarke, 1993):

\[
J = E[\sum_{j=0}^{N} e_p(t+j+k)^T e_p(t+j+k) + \sum_{j=0}^{N} \mathcal{X}_p(t+j+k)^T \mathcal{X}_p(t+j+k)] \mid t \]

where \( E[\cdot|t] \) denotes the conditional expectation, conditioned on measurements up to time \( t \) and \( \mathcal{X}_p(t) \) denotes a scalar control weighting. The future optimal control signal is to be calculated for the interval \( \tau \in [t, t+N] \).

As mentioned above the state-space models generating the signals \( r_p \) and \( y_p \) may include any dynamic cost-function weighting \( \mathcal{P}_p(z^{-1}) \), such as a low-pass filter to penalise the low-frequency disturbances. The cost-function for the problem of interest can be defined to have a similar form but with some enhancements. A term to limit the gains of the controller (denoted \( k_r(t) \)) may be added into the cost-index, so that large
gain magnitudes are penalized. The rate of gain variations \( \Delta k_i(t) \) may also be costed, where:

\[
\Delta k_i(t) = k_i(t) - k_i(t-1) \tag{25}
\]

A terminal or end-state cost-term, which represents the error in the desired end-state, can also be included to allow robustness and stability properties to be modified. This end-state error:

\[
e_i(t + i + k) = r_i(t + i + k) - x(t + i + k)
\]

where \( r_i(t + i + k) \) denotes an end-state setpoint. In some nonlinear applications, the steady-state control \( U_{ss} \), to provide a given output, is calculable, and this can be used in the criterion.

### 3.1 Optimal Control Cost-Function and Solution

The multi-step RS-GPC cost-function \( J = E[|J(t)|] \) is to be minimized that can be written in a concise vector-matrix form using the additional gain terms as follows:

\[
J = E[E_{P_{214,N}}x(E_{P_{214,N}} + (U_{N}^{0} - U_{N}^{-} \Lambda_2^2(U_{N}^{0} - U_{N}^{-})) + \Delta k_i^2(t) + e_i(t + N + k) P_i e_i(t + N + k)|t]\}

The weightings and signals being penalized include:

- The signal \( E_{P_{214,N}} \) represents the dynamically weighted future tracking error terms.
- The cost-weightings on the future inputs \( u_i \) are defined as \( \Lambda_2^2 = \text{diag}\{\lambda_2^2, \lambda_1^2, ..., \lambda_N^2\} \).
- The cost-weightings on the controller gains \( k_i(t) \) are defined as \( \Lambda_k^2 = \text{diag}\{\lambda_k^2, \lambda_k^2, ..., \lambda_k^2\} > 0 \).
- The weighting on the increment of the gains \( \Delta k_i(t) \) is \( \Lambda_\Delta^2 = \text{diag}\{\gamma_1^2, \gamma_1^2, ..., \gamma_N^2\} \) and on end-state error \( P_i \geq 0 \).

The integer \( N \) is the prediction horizon and \( N_k \) is the number of states in the low-order observer fed back through gains \( k_i(t) \). Assume a Kalman filter is used for state-estimation and prediction in the optimization (background processing), then the vector of future errors can be replaced by the orthogonal predicted errors and estimation error terms. From (26) obtain:

\[
J = E[E_{P_{214,N}}x(E_{P_{214,N}} + (U_{N}^{0} - U_{N}^{-} \Lambda_2^2(U_{N}^{0} - U_{N}^{-})) + \Delta k_i^2(t) + e_i(t + N + k) P_i e_i(t + N + k)|t]\}

\[
\Delta k_i(t) = k_i(t) - k_i(t-1). \tag{27}
\]

From (15), the end-state error can be expanded as follows:

\[
e_i(t + N + k) = r_i(t + N + k) - d_{ed}(t + N + k) - \mathcal{A}_{i}^0 \hat{x}(t + k | t)
\]

Exploiting orthogonality properties:

\[
E[|e_i(t + N + k) + \mathcal{A}_{i}^0 \hat{x}(t + k | t)|t] = \left(r_{ad}(t + k) - \mathcal{F}_{P_{214,N}}U_{N}^{0}\right) - \left(\mathcal{F}_{P_{214,N}}U_{N}^{0}\right) + \mathcal{A}_{i}^0 \hat{x}(t + k | t)
\]

The final term, that is not dependent on control action:

\[
J_f(t) = \left[E\left(\mathcal{F}_{P_{214,N}}U_{N}^{0}\mathcal{A}_{i}^0 \hat{x}(t + k | t)\right)\right] + J_f(t) \tag{28}
\]

**Cost-simplification:** The terms in the criterion can be simplified by using the orthogonality of the estimate \( \hat{e}_{P_{214,N}} \) and estimation error \( \hat{e}_{P_{214,N}} \). The vector form may be obtained, from (28), as:

\[
J = \hat{e}_{P_{214,N}} + \mathcal{A}_{i}^0 \hat{x}(t + k | t)
\]

The future error estimates, from (19), may now be written as:

\[
\hat{J}_{P_{214,N}} = \hat{J}_{P_{214,N}} + \mathcal{A}_{i}^0 \hat{x}(t + k | t)
\]

The cost-function expansion: The cost (30) may be expanded as:

\[
J = \left(U_{N}^{0} - U_{N}^{-} \Lambda_2^2(U_{N}^{0} - U_{N}^{-}) + \Delta k_i^2(t) + e_i(t + N + k) P_i e_i(t + N + k)|t]\}

\[
\Delta k_i(t) = k_i(t) - k_i(t-1). \tag{27}
\]

From (15), the end-state error can be expanded as follows:

\[
e_i(t + N + k) = r_i(t + N + k) - d_{ed}(t + N + k) - \mathcal{A}_{i}^0 \hat{x}(t + k | t)
\]

\[
\hat{J}_{P_{214,N}} + \mathcal{A}_{i}^0 \hat{x}(t + k | t)
\]

Parameterized Control Action: The expression for the parameterized control involves the \( m \)-control inputs and the \( q \) reduced-order observer state-estimates. The gain vector:

\[
\mathcal{A}_{i}^0 \hat{x}(t + k | t)
\]
Denote the vector of $q$ low order observer state-estimates as:
\[
\hat{x}_i(t) = \left[ \hat{x}_i^1, \hat{x}_i^2, \ldots, \hat{x}_i^q \right] \in \mathbb{R}^q
\]
(35)
The observer based control action, using (34) and (35):
\[
u(t) = K_c(t)\hat{x}_i(t)
\]
(36)
\[
= \begin{bmatrix}
k_{c1}\hat{x}_i^1 + k_{c2}\hat{x}_i^2 + \ldots + k_{cq}\hat{x}_i^q \\
k_{c21}\hat{x}_i^1 + k_{c22}\hat{x}_i^2 + \ldots + k_{cq2}\hat{x}_i^q \\
\ldots \\
k_{cq1}\hat{x}_i^1 + k_{cq2}\hat{x}_i^2 + \ldots + k_{cqq}\hat{x}_i^q
\end{bmatrix}
\]
\[
\hat{F}_i(t) = \begin{bmatrix}
\hat{x}_i^1(t)
\hat{x}_i^2(t)
\vdots
\hat{x}_i^q(t)
\end{bmatrix}
\]
(37)
Also, introduce a block diagonal matrix $\tilde{F}_i(t)$ as:
\[
\hat{F}_i(t) = \mathrm{diag}\left(\begin{bmatrix}
\hat{x}_i^1(t) \\
\hat{x}_i^2(t) \\
\vdots \\
\hat{x}_i^q(t)
\end{bmatrix}\right)
\]
(38)
The control signal $u(t) = \hat{F}_i(t)k_i(t)$ follows as:
\[
u(t) = \hat{F}_i(t)k_i(t)
\]
(39)
\[
= \begin{bmatrix}
k_{c1}
\vdots
k_{cq}
\end{bmatrix}
\begin{bmatrix}
\hat{x}_i^1(t)
\hat{x}_i^2(t)
\vdots
\hat{x}_i^q(t)
\end{bmatrix}
\]
\[
\hat{F}_i(t) = \begin{bmatrix}
\hat{x}_i^1(t)
\hat{x}_i^2(t)
\vdots
\hat{x}_i^q(t)
\end{bmatrix}
\]
(40)
\[
\left[\begin{array}{c}
u(t) \\
u(t+1) \\
\vdots \\
u(u(N))
\end{array}\right] = \left[\begin{array}{c}
\hat{F}_i(t) \\
\hat{F}_i(t+1) \\
\vdots \\
\hat{F}_i(t+u(N))
\end{array}\right] k_i(t)
\]
\[
= \begin{bmatrix}
k_{c1}
\vdots
k_{cq}
\end{bmatrix}
\begin{bmatrix}
\hat{x}_i^1(t)
\hat{x}_i^2(t)
\vdots
\hat{x}_i^q(t)
\end{bmatrix}
\]
\[
\hat{F}_i(t) = \begin{bmatrix}
\hat{x}_i^1(t)
\hat{x}_i^2(t)
\vdots
\hat{x}_i^q(t)
\end{bmatrix}
\]
(41)
and $\hat{x}_i^j(t+i) = \begin{bmatrix}
\hat{x}_i^1(t+i) \\
\hat{x}_i^2(t+i) \\
\vdots \\
\hat{x}_i^q(t+i)
\end{bmatrix}$.
The assumption is that the gains are constant throughout the prediction horizon and the vector of future controls may then be found given the predicted state-estimates. Let the matrix,
\[
U_{j_i}(t) = \left[\begin{array}{c}
\hat{F}_i^j(t) \\
\hat{F}_i^j(t+1) \\
\vdots \\
\hat{F}_i^j(t+u(N))
\end{array}\right]^T
\]
(42)
The i step-ahead control follows as: $u(t+i) = \hat{F}_i(t+i)k_i(t)$ or
\[
u(t+i) = \mathrm{diag}\left(\begin{bmatrix}
\hat{x}_i^1(t+i) \\
\hat{x}_i^2(t+i) \\
\vdots \\
\hat{x}_i^q(t+i)
\end{bmatrix}\right) k_i(t)
\]
(43)
The vector of future controls may now be written in terms as:
\[
U_{j_i}(t) = U_{j_i}(t)k_i(t)
\]
(44)
\[
\hat{F}_i(t) = \begin{bmatrix}
\hat{x}_i^1(t) \\
\hat{x}_i^2(t) \\
\vdots \\
\hat{x}_i^q(t)
\end{bmatrix}
\]
(45)
The time-varying matrix $\hat{A}_{j_i}(t)$ in this expression satisfies:
\[
\hat{A}_{j_i}(t) = \left(\begin{array}{c}
\hat{A}_{j_i}(t) \\
\vdots \\
\hat{A}_{j_i}(t)
\end{array}\right)
\]
(46)
Define $\hat{A}_{j_i}(t)$ and substitute from (31) to obtain:
\[
\hat{A}_{j_i}(t) = U_{j_i}(t)\hat{F}_{j_i}(t) + U_{j_i}(t)\hat{F}_{j_i}(t)
\]
(47)
\[
\hat{A}_{j_i}(t) = U_{j_i}(t)\hat{F}_{j_i}(t) + U_{j_i}(t)\hat{F}_{j_i}(t)
\]
(48)
\[
J = \begin{bmatrix}
k_{c1} & k_{c2} & \ldots & k_{cq}
\end{bmatrix}
\begin{bmatrix}
\hat{x}_i^1(t) \\
\hat{x}_i^2(t) \\
\vdots \\
\hat{x}_i^q(t)
\end{bmatrix}
\]
(49)
\[
\hat{A}_{j_i}(t) = U_{j_i}(t)\hat{F}_{j_i}(t) + U_{j_i}(t)\hat{F}_{j_i}(t)
\]
(50)
\[
\hat{A}_{j_i}(t) = U_{j_i}(t)\hat{F}_{j_i}(t) + U_{j_i}(t)\hat{F}_{j_i}(t)
\]
(51)
\[
\hat{A}_{j_i}(t) = U_{j_i}(t)\hat{F}_{j_i}(t) + U_{j_i}(t)\hat{F}_{j_i}(t)
\]
(52)
This can be simplified by noting \( \Delta k_i(t) = k_i(t) - k_i(t-1) \) where
\[
\Delta k_i(t) \Delta \tilde{A}_i(t) \Delta \tilde{A}_i(t) = k_i^T(t) \tilde{A}_i^T(t) k_i(t) - k_i^T(t-1) \tilde{A}_i^T(t-1) k_i(t) - k_i^T(t) \Delta \tilde{A}_i^T(t-1) k_i(t) + k_i^T(t-1) \Delta \tilde{A}_i^T(t-1) k_i(t)
\]
The cost expression (48) may therefore be written as:
\[
J = k_i^T(t) \tilde{D}_{\text{P}_{1+k},N} + \tilde{D}_{\text{P}_{1+k},N} \Delta \tilde{A}_i(t) - k_i^T(t-1) \Delta \tilde{A}_i(t) k_i(t) \]
\[
+ k_i^T(t) \left( U_j^T(t) \text{I}_{k+1,N} + U_j^T(t) J(t) + \Delta \tilde{A}_i^T(t-1) k_i(t) + k_i^T(t-1) \Delta \tilde{A}_i^T(t-1) k_i(t) \right) + k_i^T(t) \tilde{D}_{\text{P}_{1+k},N} \tilde{D}_{\text{P}_{1+k},N}
\]
\[
+ \rho_d^T(t+k) P_{r_d}(t+k) + U_j^T(t) \text{I}_{k+1,N} + J(t) + J_j(t)
\]
(49)

To simplify (49), define the following time-varying matrix:
\[
\mathcal{X}_{t+k,N} = U_j^T(t) \text{I}_{k+1,N} + J(t) + J_j(t)
\]

Cost-function: Substituting these results into (49):
\[
J = k_i^T(t) \left( \tilde{D}_{\text{P}_{1+k},N} - \tilde{A}_i^T(t) \right) + k_i^T(t) \left( \tilde{D}_{\text{P}_{1+k},N} - \tilde{A}_i^T(t-1) \right) k_i(t)
\]
\[
+ k_i^T(t) \left( \tilde{D}_{\text{P}_{1+k},N} \Delta \tilde{A}_i(t) + \tilde{D}_{\text{P}_{1+k},N} \Delta \tilde{A}_i(t-1) k_i(t) + k_i^T(t-1) \Delta \tilde{A}_i^T(t-1) k_i(t) \right)
\]
\[
+ \rho_d^T(t+k) P_{r_d}(t+k) + U_j^T(t) \text{I}_{k+1,N} + J(t) + J_j(t)
\]
(51)

The terms within the square brackets are independent of the control action. The necessary condition for optimality requires the gradient of the cost-function to be set to zero, to obtain the vector of future optimal controls. The optimal gain:
\[
k_i(t) = -\mathcal{X}_{t+k,N}^{-1} \left( \tilde{D}_{\text{P}_{1+k},N} - \tilde{A}_i^T(t-1) \right)
\]
(52)

The sufficient condition requires that the second-derivative or Hessian matrix be positive definite \( \mathcal{X}_{t+k,N} > 0 \), which is ensured because the control gain is positive definite.

Asymptotic Behaviour: From (52) if the rate of change cost term \( \tilde{A}_i \rightarrow \infty \) the limiting gain \( k_i(t) = k_i(t-1) \) becomes constant.

Minimum cost: Substituting in (51) for the gain \( k_i(t) \) in (52) and simplifying, the minimum-cost becomes:
\[
J_{\text{sw}} = \tilde{D}_{\text{P}_{1+k},N} \tilde{D}_{\text{P}_{1+k},N} - \left( \tilde{D}_{\text{P}_{1+k},N} - k_i^T(t-1) \tilde{A}_i \right) \Delta \tilde{A}_i^T(t-1) k_i(t)
\]
\[
+ k_i^T(t-1) \tilde{A}_i^T(t-1) k_i(t) + \rho_d^T(t+k) P_{r_d}(t+k) + U_j^T(t) \text{I}_{k+1,N} + J(t) + J_j(t)
\]
where \( \tilde{D}_{\text{P}_{1+k},N} \) was defined in (47) and gain \( k_i(t-1) \) is known.

Sub-optimality: The minimum-cost (53) may be compared with the minimum for conventional GPC to minimize (24), when the weightings \( \tilde{A}_i \), \( \tilde{A}_i \) and \( P_i \) tends to zero, so from (29) \( J_i(t) \rightarrow 0 \) \( \mathcal{X}_{t+k,N} \rightarrow U_j^T(t) \text{I}_{k+1,N} + J_j(t) \) and \( \tilde{D}_{\text{P}_{1+k},N} \rightarrow C_{t+k,N} \tilde{D}_{\text{P}_{1+k},N} \). Assuming \( U_j^T(t) \) is zero and that \( \mathcal{X}_{t+k,N} \) has an inverse the minimal cost, from (53) becomes:
\[
J_{\text{sw}} \rightarrow \tilde{D}_{\text{P}_{1+k},N} \tilde{D}_{\text{P}_{1+k},N} - \tilde{D}_{\text{P}_{1+k},N} \mathcal{X}_{t+k,N}^{-1} \tilde{D}_{\text{P}_{1+k},N} \mathcal{X}_{t+k,N} + J_j(t)
\]

This minimum cost will be larger relative to the minimal-cost in the equivalent unrestricted GPC control problem, given as:
\[
J_{\text{sw}} = \tilde{D}_{\text{P}_{1+k},N} \left( I - V_{t+k,N} \right) \tilde{D}_{\text{P}_{1+k},N} + \tilde{A}_i^T(t) V_{t+k,N} + \tilde{A}_i^T(t) \tilde{D}_{\text{P}_{1+k},N} + J_j(t)
\]
(55)

The two costs (54) and (55) approach the same value in the special case when \( U_j^T(t) \) is square and full rank.

4.1 RS-GPC Main Theorem

Theorem 1: Observer Based RS-GPC Control

Consider the system and assumptions given at the start of §2. The observer based restricted-structure generalized predictive controller is required to minimize the cost-function:
\[
J = E_P q_{P_{t+k,N}} \left( q_{P_{t+k,N}} - U_j^T(t-1) \right) \Delta \tilde{A}_i(t) k_i(t) + \Delta \tilde{A}_i^T(t) \tilde{A}_i \Delta \tilde{A}_i(t) k_i(t) + \mathcal{X}_{t+k,N}^{-1} \left( \tilde{D}_{\text{P}_{1+k},N} - \tilde{A}_i \right) \Delta \tilde{A}_i^T(t-1) k_i(t)
\]
\[
+ \rho_d^T(t+k) P_{r_d}(t+k) + U_j^T(t) \text{I}_{k+1,N} + J_j(t) + J(t)
\]
(56)

The RS-GPC control is parameterised as \( u(t) = \hat{F}_i(t) k_i(t) \) where \( \hat{F}_i(t) \) is given by (41). The optimal gains found to minimize (56) using a receding horizon philosophy:
\[
k_i(t) = -\mathcal{X}_{t+k,N}^{-1} \left( \tilde{D}_{\text{P}_{1+k},N} - \tilde{A}_i \right) k_i(t-1)
\]
\[
= -\mathcal{X}_{t+k,N}^{-1} \left( \tilde{D}_{\text{P}_{1+k},N}^0 - \tilde{A}_i \right) k_i(t+k-1)
\]
\[
-\mathcal{X}_{t+k,N}^{-1} \left( \tilde{D}_{\text{P}_{1+k},N} - \tilde{A}_i \right) k_i(t)
\]
(57)

The matrix \( \mathcal{X}_{t+k,N} \) is given by (50) and \( \mathcal{P}_{t+k,N}, \mathcal{C}_{t+k,N} \) and \( \tilde{D}_{\text{P}_{1+k},N} \) by (46). The gain-vector \( k_i(t) \) has the form:
\[
k_i(t) = \left[ \begin{array}{c} k_{1T} \\
\vdots \\
k_{iT} \\
\vdots \\
k_{mT}
\end{array} \right]
\]

The optimal control is given by (57) and parameterized vector of future controls for the prediction error computations:
\[
u_{t+k,N} = U_{j+k}(t) k_i(t)
\]
(59)

Solution: These results were derived in the solution presented before the theorem.

Comments on the control solution: The “denominator matrix” in the RS-GPC controller gain expression (58) is full rank because of the cost and system descriptions. It is interesting that the expression for the gain-vector is of a similar form to the usual GPC solution but the denominator matrix in (58) can be of much lower dimension than arises in GPC control. The size of this matrix depends upon the number of gains in the parameterised controller, which in turn depends upon the order of the observer used in the feedback loop. The weightings \( \tilde{A}_i \) and \( \tilde{A}_i \) ensures that the denominator matrix does not become singular.
4.2 Hard and Soft Constraints in RS-GPC

Soft constraints can be applied to the controller gains, or to the rate of change of gains, by changing the weighting terms \( \lambda^1_k \) and \( \lambda^2_k \) in (56). Hard constraints may be applied in almost the same way as for GPC algorithms using Quadratic Programming (QP). Constraints may be applied to the magnitude of the controller gains, or to their rate of change. They may also be applied to the control signal inputs and outputs. The ability to bound the gain or gain variations is essential in some applications. Limiting controller gains has a direct effect on power demands and an indirect effect on nonlinear system behaviour and stability.

4.3 Low Order Observer Design

The low-order observer within the loop is not the same as the Kalman-filter in the background processing. It is chosen to be low-order to improve robustness and so that it can be approximated by a scheduled solution for real-time implementation. The design of the observer is important but not critical to the performance obtained. This is because the optimization process computes gains that optimize the solution whatever the choice of observer. Clearly if there is a perverse choice of observer, the optimal control will find it harder to provide a “best” solution. The observer should therefore be designed using a rational philosophy but the solution may not be very sensitive. The observer can be found by basing the plant model on just the low frequency plant modes or it can be model reduced and a low order Kalman-filter can be computed. If the state-space model is linear time-invariant, a Luenberger-observer can be designed. There is an interesting extension suggested by Morari and Maeder (2012) for nonlinear predictive controls where the plant model is augmented by a disturbance model, which accounts for model mismatch and disturbances. They noted, “In principle it would be possible to use an elaborate nonlinear model in the observer and a simple (locally linearized) model in the controller as long as the steady-state characteristics match.” The opportunity to improve robustness and achieve offset free tracking in the RS-GPC controller is clear.

4.4 Cost-Function Tuning Variables

Tuning the controller should be as simple as traditional MPC (Grimble and Majecki, 2010). The weighting on the control signal provides a simple way to vary the speed of response. The initial choice of weighting on the error signal can be taken as the identity or if integral action is included then the weighting gain can be scaled to the identity. The remaining weightings \( F_{c_1} = \lambda^1_k \) and \( F_{c_2} = \lambda^2_k \) are on the magnitude of the gains and the rate of change of gains, and it is useful to limit these gains.

5 ROLLING MILL LOOPER CONTROL DESIGN

The interstand looper involves a roll on a torque-controlled arm located between the adjacent stands of a multi-stand hot rolling mill. It maintains a constant tension between the stands to correct any disturbances in the strip mass flow, during acceleration and deceleration. The plant I/O structure is shown in Fig. 3 and the problem is described in more detail in Hearns et al., (1996).

![Fig. 3: Looper Control Plant Model](image)

![Fig. 4: Comparison Reduced-Order Observer Based RS-GPC with MPC Setpoint Following (Observer reduced by a single state)](image)

The results in Fig. 4 compare model predictive control (MPC) and the RS-GPC algorithm, aimed at good looper-angle reference tracking. The observer is one state lower than full-order and is a suboptimal Kalman Filter within the loop. The Fig. 5 shows the observer based RS-GPC solution when the observer uses a plant model, which is five or six states lower than the full-order case and using the same cost-function. It is surprising that tracking performance is so good for this case but recall the background processing which involves the predictive control algorithm computes the “optimal” solution and attempts to compensate for the use of the low-order observer. The variances of tracking errors can be considered but they are not very meaningful, since the individual variances are not being minimized and MPC and RS cost-functions are very different.

<table>
<thead>
<tr>
<th></th>
<th>Looper Angle</th>
<th>Strip Tension</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPC</td>
<td>5776.2</td>
<td>15.91</td>
</tr>
<tr>
<td>RS-GPC</td>
<td>7110.7</td>
<td>0.25</td>
</tr>
<tr>
<td>RS -5s</td>
<td>6343.5</td>
<td>0.17</td>
</tr>
<tr>
<td>RS -6s</td>
<td>6620.1</td>
<td>0.24</td>
</tr>
</tbody>
</table>

The RS results are for the observer model reduced by one, five and six states, respectively. However, the RS criterion has the term \( k^1(t) \lambda^1_k k^1(t) \) that depends on the number of states. The higher-order observer RS control includes five more weighting terms than the RS -6 states case so again it is difficult to compare results in this way.
Fig. 5: Comparison When Observer States Reduced by 5 or 6 States

Note that if the low-order observer is made very fast by reducing the assumed measurement noise variance the RS results are all rather similar. As noted above the background processing can therefore ensure high performance is achieved providing the assumed low-order observer does not slow down the system too markedly.

6 CONCLUDING REMARKS

When a low order observer is to be used in a feedback control, there is not normally an exact analytic solution to the resulting optimal control problem. The main contribution is that the proposed predictive control solution here does not involve any approximation. The RS-GPC approach has potential in automotive applications such as Majecki et al. (2015, 2017) where the RS structure can be simplified for implementation. A low-order controller also seems to have a mother nature intended natural robustness. Moreover, predictive controls often exhibit an improvement in robustness over long prediction-horizons and this property has been observed in application examples.

The controller has natural feed-forward terms and provides some measure of time delay compensation. The use of an optimisation method to compute the parameterised controller gains provides the benefits of model-based control and reduces the tuning process to the selection of simple cost-function weightings. The advances on nonlinear predictive control (Findeisen et al. 2003, Mayne et al., 2005, 2014) suggests a number of extensions.

REFERENCES


