NONLINEAR PREDICTIVE CONTROL FOR MANUFACTURING AND ROBOTIC APPLICATIONS

PROF. MIKE J. GRIMBLE AND DR. ANDRZEJ W. ORDYS

Industrial Control Centre, University of Strathclyde,
50 George Street, Glasgow G1 1QE, UK,
m.grimble@eee.strath.ac.uk, a.ordys@eee.strath.ac.uk

Abstract:
The paper discusses predictive control algorithms in the context of applications to robotics and manufacturing systems. Special features of such systems, as compared to traditional process control applications, require that the algorithms are capable of dealing with faster dynamics, more significant unstabilities and more significant contribution of non-linearities to the system performance. The paper presents the general framework for state-space design of predictive algorithms. Linear algorithms are introduced first, then, the attention moves to non-linear systems. Methods of predictive control are presented which are based on the state-dependent state space system description. Those are illustrated on examples of rather difficult mechanical systems.

Keywords: Model Based Predictive Control, Non-linear systems, State-Dependent Riccati Equations, Linear Quadratic Gaussian Control, Optimization.

1. INTRODUCTION

It is well known that the Model Based Predictive Control (MBPC) originated in process industry where it was applied to slow processes which could be adequately described by rather crude linear models. The advent in computational algorithms and real time control hardware stimulates attempts to transfer this method to more demanding applications. In this paper we aim to present some developments in predictive control which are relevant for manufacturing and robotic applications.

The paper is organized as follows:

In section 2 we deal with linear predictive control. After introducing the notation and defining the control structure the method of incorporating the non-zero reference signal into the state-space equations is presented. Based on this, the state-space Generalized Predictive Controller (GPC) is derived and next the stability of predictive schemes is discussed. Finally, a Linear Quadratic Gaussian Predictive Controller (LQGPC) is introduced. This algorithm can improve stability by incorporating finite horizon tuning parameters into infinite horizon optimization. Also, it proves to provide better performance than other predictive schemes, perhaps due to the fact that the values of tuning parameters which would result in instability for, say GPC, are still within stable region for LQGPC. It is important for fast mechanical systems where good tracking performance must be achieved. This algorithm can be computationally involving therefore some simplifications are presented which enable more efficient calculations of controls.

Section 3 provides a short overview of three selected methods of non-linear predictive control. The first of them is contractive predictive control. Here, a special contractive constraint imposed on the state of the system at the end of the prediction horizon will assure stability. At the same time, within the prediction horizon, the control moves are hoped to have enough freedom to provide good performance.

The second is the approach developed by researchers from Oxford University. The emphasis here is on feasibility for real time computations. The method is based on invariant ellipsoidal sets. Finally, the last method presented in this section uses fuzzy Takagi-Sugeno models for prediction.

In section 4 we concentrate on algorithms which are inspired by the idea of global linearization for non-linear systems. The formulation of so-called state dependent state-space equations has lead to development of a technique which borrows from algorithmic solution for a standard linear quadratic problem and therefore is called State Dependent Riccati Equations (SDRE). This method is applied within the framework of predictive control for both finite horizon (GPC) and infinite horizon (LQGPC) algorithms. We notice that in the predictive schemes the future control actions within a finite horizon are predicted and available at a current time. This fact leads to improvements in defining the state-dependent system model and therefore to improvements in the accuracy of control.

Section 5 shows examples of application of the state-dependent predictive algorithms. This is a very recent research direction and the results are not conclusive yet. However, the numerical examples presented are very encouraging.

In section 6 we conclude the paper trying to predict the main future research directions and main obstacles to be overcome.

2. LINEAR PREDICTIVE CONTROL

2.1. Problem statement

The linear, time-invariant, discrete-time, finite-dimensional, $n_y \times n_u$ multivariable system of interest is represented in state equation form and is assumed to be stabilizable and detectable. The subsystem S_1 denotes the plant model and the reference signal is assumed to be generated by the subsystem S_0. A slight generalization of the problem is to consider the inferred outputs $\{y_h(t)\}$, rather than the plant outputs $\{y(t)\}$, as being the signals to be controlled. The system model and the state-space plant equations to be considered, are therefore of the form:

State :

$$x_1(t+1)=A_1x_1(t)+B_1u(t)+D_1\xi_1(t) \qquad (1)$$

Output :

$$y_1(t)=C_1x_1(t) \qquad (2)$$

Observations :

$$z_1(t)=y_1(t)+v_1(t) \qquad (3)$$

Inferred output :

$$y_h(t)=H_1x_1(t) \qquad (4)$$

and the state $x_1(t)\in R^{n_1}$, output $y_1(t)\in R^{n_y}$, control $u(t)\in R^{n_u}$, inferred output $y_h(t)\in R^{n_h}$, disturbances $\xi_1(t)\in R^{q_1}$ and noise $v_1(t)\in R^{n_y}$. The zero-mean, white noise signals $\{v_1(t)\}$ and $\{\xi_1(t)\}$ have the following covariance matrices:

$cov[v_1(t),v_1(\tau)]=R_{v_1}\delta_{t\tau}>0$ and

$cov[\xi_1(t),\xi_1(\tau)]=I_{q_1}\delta_{t\tau}$

where the cross-covariances are assumed to be null.

A model is required to predict the future values of the inferred output signal. From equations (1) and (4) obtain:

$$\begin{bmatrix} y_h(t+1) \\ y_h(t+2) \\ \vdots \\ y_h(t+N) \end{bmatrix} = \begin{bmatrix} H_1A_1 \\ H_1A_1^2 \\ \vdots \\ H_1A_1^N \end{bmatrix} x_1(t)$$
$$+\begin{bmatrix} H_1B_1 & 0 & \cdots & 0 \\ H_1A_1B_1 & H_1B_1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ H_1A_1^{N-1}B_1 & H_1A_1^{N-2}B_1 & \cdots & H_1B_1 \end{bmatrix}\begin{bmatrix} u(t) \\ u(t+1) \\ \vdots \\ u(t+N-1) \end{bmatrix}$$
$$+\begin{bmatrix} H_1D_1 & 0 & \cdots & 0 \\ H_1A_1D_1 & H_1D_1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ H_1A_1^{N-1}D_1 & H_1A_1^{N-2}D_1 & \cdots & H_1D_1 \end{bmatrix}\begin{bmatrix} \xi_1(t) \\ \xi_1(t+1) \\ \vdots \\ \xi_1(t+N-1) \end{bmatrix} \qquad (5)$$

With an obvious definition of terms this equation for the inferred outputs, may be written in the more concise form:

$$Y^h_{t+1,N}=H_Nx_1(t)+G_NU_{t,N}+N_NW_{t,N} \qquad (6)$$

The assumptions on the reference signal are similar to those presented by Tomizuka and co-authors [51], [53]. The reference signal $\{r_h(t)\}$ will be generated by the asymptotically stable linear stochastic state equation system model:

$$x_{r0}(t+1)=A_rx_{r0}(t)+D_r\xi_0(t) \qquad (7)$$

The zero-mean white-noise source $\{\xi_0(t)\}$ is assumed to have the unity covariance. The desired future value of the reference signal *p* steps-ahead, is defined as a linear function of the current "reference state":

$$r_h(t+p)=H_rx_{r0}(t) \qquad (8)$$

where $p\geq N\geq 1$ is greater than or equal to the number of steps in the output prediction. New state equation variables may be defined, when $p > 1$, as delayed values of the reference signal:

$$\begin{bmatrix} x_{r0}(t+1) \\ x_{r1}(t+1) \\ x_{r2}(t+1) \\ \vdots \\ x_{r(p-1)}(t+1) \end{bmatrix} = \begin{bmatrix} x_{r0}(t+1) \\ r_h(t+p) \\ r_h(t+p-1) \\ \vdots \\ r_h(t+2) \end{bmatrix} =$$

$$\begin{bmatrix} A_r & 0 & \cdots & & & 0 \\ H_r & 0 & \cdots & & & \\ 0 & I & & & & \\ \vdots & & \ddots & & & 0 \\ 0 & 0 & \cdots & 0 & I & 0 \end{bmatrix} \begin{bmatrix} x_{r0}(t) \\ x_{r1}(t) \\ x_{r2}(t) \\ \vdots \\ x_{r(p-1)}(t) \end{bmatrix} + \begin{bmatrix} D_r \\ 0 \\ \vdots \\ 0 \end{bmatrix} \xi_0(t) \tag{9}$$

which may be written, with an obvious definition of terms, in the vector form:

$$x_0(t+1) = A_0 x_0(t) + D_0 \xi_0(t) \tag{10}$$

The current and future reference values can then be obtained as:

$$\begin{bmatrix} r_h(t+1) \\ r_h(t+2) \\ \vdots \\ r_h(t+p-1) \\ r_h(t+p) \end{bmatrix} = \begin{bmatrix} 0 & 0 & & \cdots & 0 & I \\ 0 & 0 & & \cdots & I & 0 \\ \vdots & \vdots & & \cdot^{\cdot^{\cdot}} & & \\ 0 & I & 0 & \cdots & 0 & 0 \\ H_r & 0 & 0 & \cdots & & 0 \end{bmatrix} \begin{bmatrix} x_{r0}(t) \\ x_{r1}(t) \\ \vdots \\ \\ x_{r(p-1)}(t) \end{bmatrix} \tag{11}$$

which can be written in the form:

$$R_p(t+1) = H_p x_0(t) \tag{12}$$

The N $(1 \le N \le p)$ future set-point or reference values in the cost-function can be denoted as:

$$R_N(t+1) = H_{RN} H_p x_0(t) = C_0 x_0(t) \tag{13}$$

The equations for the total system will now be obtained and these will determine the size of the Riccati equations in the control and estimation problems. Combining the state equations for the reference and the plant obtain the total augmented system as:

$$\begin{bmatrix} x_0(t+1) \\ x_1(t+1) \end{bmatrix} = \begin{bmatrix} A_0 & 0 \\ 0 & A_1 \end{bmatrix} \begin{bmatrix} x_0(t) \\ x_1(t) \end{bmatrix} + \begin{bmatrix} 0 \\ B_1 \end{bmatrix} u(t) + \begin{bmatrix} D_0 \xi_0(t) \\ D_1 \xi_1(t) \end{bmatrix} \tag{14}$$

which may be written in the more concise form:

$$X(t+1) = AX(t) + Bu(t) + D\xi(t) \tag{15}$$

The equation for the future inferred outputs to be costed in the criterion may be written in terms of the state vector *X(t)*, using (5), as:

$$Y^h_{t+1,N} = \begin{bmatrix} 0 & H_N \end{bmatrix} X(t) + G_N U_{t,N} + N_N W_{t,N} \tag{16}$$

and from the reference vector equation (13):

$$R_{t+1,N} = \begin{bmatrix} C_0 & 0 \end{bmatrix} X(t) \tag{17}$$

The error signal may now be written, using these two equations, as:

$$\begin{aligned} E_{t+1,N} &= R_{t+1,N} - Y^h_{t+1,N} \\ &= \begin{bmatrix} C_0 & -H_N \end{bmatrix} X(t) - G_N U_{t,N} - N_N W_{t,N} \end{aligned} \tag{18}$$

That is, the predicted future error term that will appear in the cost-function is:

$$E_{t+1,N} = \tilde{H} X(t) + \tilde{G} U_{t,N} + \tilde{W}_{t,N} \tag{19}$$

where

$$\tilde{H} = \begin{bmatrix} C_0 & -H_N \end{bmatrix},\ \tilde{G} = -G_N,\ \tilde{W}_{t,N} = -N_N W_{t,N} \tag{20}$$

2.2. State-space GPC algorithm

The full derivation of GPC controller in state-space form may be found in [38]. Only the main points are recalled below.

The performance index to be minimized is defined as follows:

$$\begin{aligned} J_t = E\Bigg\{ \sum_{j=0}^{N} \Big[\big(r_h(t+j+1) - y_h(t+j+1)\big)^T Q_j \big(r_h(t+j+1) + \\ - y_h(t+j+1)\big) \Big] + u(t+j)^T R_j u(t+j) \Bigg\} \end{aligned} \tag{21}$$

where $E\{.\}$ denotes the unconditional expectation operator, and the error Q_j and control R_j weightings can be different for future steps *j*.

The cost-function can be simplified by introducing the block diagonal weighting matrices $\tilde{Q} = \mathrm{diag}\{Q_1,...,Q_N\}$ and $\tilde{R} = \mathrm{diag}\{R_0,...,R_{N-1}\}$.

The J_t term can therefore be written:

$$J_t = E\left\{ \left(R_{t+1,N} - Y^h_{t+1,N}\right)^T \tilde{Q} \left(R_{t+1,N} - Y^h_{t+1,N}\right) + U^T_{t,N} \tilde{R} U_{t,N} \right\} \tag{22}$$

Next, equations (18) and (19) are used. Note from (5) that the disturbance model $\tilde{W}_{t,N}$ includes current and future values of the white noise disturbance signal. The vector of future predictive values of the signal $U_{t,N}$ is to be calculated at time *t* and cannot utilise knowledge of future disturbance signal components. Thus, $(\tilde{H}X(t) + \tilde{G}U_{t,N})$ and $\tilde{W}_{t,N}$ are statistically independent and zero mean, and from (22):

$$\begin{aligned} J_t = E\Big\{ \left(\tilde{H}X(t) + \tilde{G}U_{t,N}\right)^T \tilde{Q} \left(\tilde{H}X(t) + \tilde{G}U_{t,N}\right) \\ + U^T_{t,N} \tilde{R} U_{t,N} \Big\} + J_0 \end{aligned} \tag{23}$$

and

$J_0 = E\{(\tilde{W}_{t,N}^T \tilde{Q} \tilde{W}_{t,N})\}$

Performing conditional expectation operation (as in standard LQG problem) the performance index, neglecting a constant (control independent) term, can be expressed as:

$$J_t = \left(\tilde{H}\hat{X}(t) + \tilde{G}U_{t,N}\right)^T \tilde{Q}\left(\tilde{H}\hat{X}(t) + \tilde{G}U_{t,N}\right) + U_{t,N}{}^T \tilde{R} U_{t,N} \quad (24)$$

where $\hat{X}(t)$ denotes the estimate of the extended state. Thus, by finding the stationary point, the vector of optimal control signals becomes:

$$U_{t,N} = -\left(\tilde{G}^T \tilde{Q} \tilde{G} + \tilde{R}\right)^{-1} \tilde{G}^T \tilde{Q} \tilde{H} \hat{X} \quad (25)$$

Equation (25) can be represented in a more familiar form when the matrix $\tilde{H}$ and the extended state $\tilde{X}$ are substituted by their appropriate definitions (20) and (14):

$$U_{t,N} = -\left(G_N{}^T \tilde{Q} G_N + \tilde{R}\right)^{-1} \left(-G_N{}^T \tilde{Q} \begin{bmatrix} C_0 & -H_N \end{bmatrix} \begin{bmatrix} \hat{x}_0(t) \\ \hat{x}_1(t) \end{bmatrix}\right) = \left(G_N{}^T \tilde{Q} G_N + \tilde{R}\right)^{-1} G_N{}^T \tilde{Q}\left(R_{t,N} - H_N \hat{x}_1(t)\right) \quad (26)$$

As predictive control is based on the receding horizon philosophy, only the first element from the vector $U_{t,N}$ is used and the optimization is performed again in the next step with (possibly) a new value of the reference signal.

2.3. Stability improvements through use the of terminal constraints or terminal cost

The state-space version of the GPC controller presented above is well know to lack guaranteed stability properties. Several authors have suggested modifications and improvements to the problem formulation which will lead to better stability of the closed loop system. A popular approach is to consider terminal constraints which could be equality constraints [9, 34], non-equality constraints [6] or terminal penalty in the cost function [1, 33]. In the latter case, if zero reference signal is assumed, the performance index (21) will change to:

$$J_t = E\left\{\sum_{j=0}^{N-1}\left[x_1(t+j+1)^T H_1^T Q_j H_1 x_1(t+j+1)\right] + u(t+j)^T R_j u(t+j)\right\} + x_1(t+N+1)^T P_{N+1} x_1(t+N+1) \quad (27)$$

where P_{N+1} represents the terminal cost weighting matrix. Using the Principle of Optimality, P_{N+1} can be selected as the cost associated with infinite horizon performance index, therefore guaranteeing that the solution has the same stability properties as the infinite horizon LQG problem.

2.4. Linear Quadratic Gaussian Predictive Control (LQGPC)

The approach, which we propose here is related to the terminal cost approach described earlier. The dynamic performance index to be minimized is defined as:

$$J = E\left\{\lim_{T\to\infty} \frac{1}{2T} \sum_{t=-T}^{T} J_t\right\} \quad (28)$$

where J_t is described by equation (21) or (22). This new performance index (28) can be considered as the previous performance index (22) plus the terminal constraint which is an infinite horizon quadratic performance index.

To be able to take into account the dynamic nature of the problem a slight modification to the state space equation of the system (15) is made:

$$X(t+1) = AX(t) + \beta U_{t,N} + D\xi(t) \quad (29)$$

where the block matrix β is constructed as follows:

$$\beta^T = \begin{bmatrix} B^T & O & \ldots & O \end{bmatrix} \quad (30)$$

Therefore, equations (29) and (15) represent exactly the same dynamic system. The modification introduced in equation (29) will enable us to substitute directly the state equation into the performance index and therefore to solve the infinite horizon dynamic optimization problem.

The cost-function term J_t may be expanded using equation (24):

$$J_t = E\Big\{X^T(t)\tilde{H}^T \tilde{Q} \tilde{H} X(t) + U_{t,N}^T(\tilde{G}^T \tilde{Q} \tilde{G} + \tilde{R})U_{t,N} + U_{t,N}^T \tilde{G}^T \tilde{Q} \tilde{H} X(t) + X^T(t) \tilde{H}^T \tilde{Q} \tilde{G} U_{t,N}\Big\} + J_0 \quad (31)$$

This cost term may therefore be written in the form:

$$J_t = J_0 + E\left\{X^T(t) Q_c X(t) + U_{t,N}^T R_c U_{t,N} + 2X^T(t) G_c U_{t,N}\right\} \quad (32)$$

where the weightings:

$Q_c = \tilde{H}^T \tilde{Q} \tilde{H}$, $R_c = \tilde{G}^T \tilde{Q} \tilde{G} + \tilde{R}$ and $G_c = \tilde{H}^T \tilde{Q} \tilde{G}$

include the cross-product term G_c.

The optimal control solution is then given by:

$$U_{t,N} = -\left(\tilde{R} + \tilde{G}^T\tilde{Q}\tilde{G} + B^T\tilde{S}B\right)^{-1}\left(\tilde{G}^T\tilde{Q}\tilde{H} + B^T\tilde{S}A\right)X(t) \tag{33}$$

where $\tilde{S}$ is a steady state solution of the Riccati equation:

$$\tilde{S}_j = \tilde{H}^T\tilde{Q}\tilde{H} + A^T\tilde{S}_{j+1}A - \left(\tilde{H}^T\tilde{Q}\tilde{G} + A^T\tilde{S}_{j+1}B\right) \times\left(\tilde{R} + \tilde{G}^T\tilde{Q}\tilde{G} + B^T\tilde{S}_{j+1}B\right)^{-1} \times\left(G^T\tilde{Q}\tilde{H} + B^T\tilde{S}_{j+1}A\right) \tag{34}$$

with the terminal condition: $\tilde{S}_{T+1} = O$. Equations (33) and (34) may be further simplified. Assuming that matrix $\tilde{S}_j$ is divided into four matrix blocks of appropriate dimensions:

$$\tilde{S}_j = \begin{bmatrix} \tilde{S}_j^1 & \tilde{S}_j^2 \\ \tilde{S}_j^{2T} & \tilde{S}_j^3 \end{bmatrix} \tag{35}$$

and using definitions of matrices as given in equations (14) and (20) obtains:

$$U_{t,N} = -\left(R + G_N{}^T\tilde{Q}G_N + B_1{}^T\tilde{S}^1B_1\right)^{-1} \left[\left(G_N{}^T\tilde{Q}G_N A_1 + B_1{}^T\tilde{S}^1A_1\right)\hat{X}(t) + \left(B_1{}^T\tilde{S}^2A_0 - G_N{}^T\tilde{Q}\right)R_N(t)\right] \tag{36}$$

and equation (34) may be split into two equations:

$$\tilde{S}_j^1 = A_1{}^T\left(H_N{}^T\tilde{Q}H_N + \tilde{S}_{j+1}^1\right)A_1 + -A_1{}^T\left(H_N{}^T\tilde{Q}G_N + \tilde{S}_{j+1}^1B_1\right)\left(\tilde{R} + G_N{}^T\tilde{Q}G_N + B_1{}^T\tilde{S}_{j+1}^1B_1\right)^{-1} \times\left(G_N{}^T\tilde{Q}H_N + B_1{}^T\tilde{S}_{j+1}^1\right)A_1 \tag{37}$$

$$\tilde{S}_j^2 = -A_1{}^T H_N{}^T\tilde{Q} + A_1{}^T\tilde{S}_{j+1}^2A_0 + -A_1{}^T\left(H^T\tilde{Q}G_N + \tilde{S}_{j+1}^1B_1\right)\left(\tilde{R} + G_N{}^T\tilde{Q}G_N + B_1{}^T\tilde{S}_{j+1}^1B_1\right)^{-1} \times\left(B_1{}^T\tilde{S}_{j+1}^2A_0 - G_N{}^T\tilde{Q}\right) \tag{38}$$

2.5. Implementation to Manufacturing Systems

A property of the above particular minimization problem enables the solution procedure to be simplified and made numerically efficient. First note that the vector of future controls, $U_{t,N}$ can be partitioned into those determining the control input *u(t)*, at time *t*, and the controls for $\tau > t$ that will now be denoted as $U_{t,N}^f$. Assuming for the moment that states may be measured, this result enables *u(t)* and $U_{t,N}^f$ to be computed separately. To demonstrate this result we will partition G_N, R_c and G_c compatibly with the partition of $U_{t,N}$. Consider the cost term *I(t)* where:

$$\begin{aligned} I(t) &= X^T(t)Q_cX(t) + U_{t,N}^TR_cU_{t,N}(t) + 2X^T(t)G_cU_{t,N} \\ &= X^TQ_cX + u^TR_{c1}u + 2X^TG_{c1}u + U_{t,N}^{fT}R_{c2}U_{t,N}^f \\ &\quad + 2u^TR_{c3}U_{t,N}^f + 2X^TG_{c2}U_{t,N}^f \end{aligned} \tag{39}$$

From (30), the dynamics of the system, involved in the state model, do not depend upon $U_{t,N}^f$. It follows that the optimization of the cost, depending upon $U_{t,N}^f$ is a finite dimensional problem. To obtain the gradient, with respect to $U_{t,N}^f$ we consider:

$$\begin{aligned} &\frac{\partial}{\partial U_{t,N}^f}((U_{t,N}^{fT}R_{c2} + 2u^T(t)R_{c3} + 2X^T(t)G_{c2})U_{t,N}^f) \\ &= 2(U_{t,N}^{fT}R_{c2} + u^T(t)R_{c3} + X^T(t)G_{c2}) \end{aligned} \tag{40}$$

Setting the gradient to zero, to obtain the optimum cost, we find the vector of future controls as:

$$U_{t,N}^f = -R_{c2}^{-1}(R_{c3}^Tu(t) + G_{c2}^TX(t)) \tag{41}$$

This provides the solution for future controls and the main problem then becomes the calculation of the feedback control at time *t*. The control at time *t* can be found by minimization of E{J_t} defined as:

$$\begin{aligned} E\{J_t\} &= E\{X^T(t)\bar{Q}_cX(t) + u^T(t)\bar{R}_cu(t) \\ &\quad + 2X^T(t)\bar{G}_cu(t)\} + J_0 \end{aligned} \tag{42}$$

where

$$\begin{aligned} \bar{Q}_c &= Q_c - G_{c2}R_{c2}^{-1}G_{c2}^T, \\ \bar{R}_c &= R_{c1} - R_{c3}R_{c2}^{-1}R_{c3}^T, \\ \bar{G}_c &= G_{c1} - G_{c2}R_{c2}^{-1}R_{c3}^T \end{aligned} \tag{43}$$

Note that when states are not available *X(t)* can be replaced by the optimal estimate $\hat{X}(t)$.

Therefore, the solution of the LQGPC optimal control problem, with the multi-step cost index may be found in two stages.

Stage 1 : Vector of future controls

$$U_{t,N}^f = -R_{c2}^{-1}(R_{c3}^Tu(t) + G_{c2}^TX(t)) \tag{44}$$

Stage 2 : Minimization in terms of current control. This minimization can be performed using polynomial description of the system as follows:

The predictive control criterion to be minimized is defined to have the following steady-state (infinite time) form:

$$J = \lim_{T \to \infty} \frac{1}{2T} E\left\{ \sum_{t=-T}^{T} \left\{ X^T(t)\bar{Q}_c X(t) + u^T(t)\bar{R}_c u(t) \right.\right.$$
$$\left.\left. + X^T(t)\bar{G}_c u(t) + u^T(t)\bar{G}_c^T X(t) \right\}\right\}$$
$$= \frac{1}{2\pi j} \oint_{|z|=1} \left\{ trace\left\{ \bar{Q}_c \Phi_{XX}(z^{-1}) + 2\bar{G}_c \Phi_{uX}(z^{-1}) \right\} \right.$$
$$\left. + trace\left\{ \bar{R}_c \Phi_{uu}(z^{-1}) \right\} \right\} \frac{dz}{z} \qquad (45)$$

The optimal control, tracking and feedback components, may be computed using:

$$u(t) = -K_0(z^{-1})R_N(t) - K_1(z^{-1})z_1(t) \qquad (46)$$

where the matrix polynomials $K_0(z^{-1})$ and $K_1(z^{-1})$ are obtained through standard minimization procedures applied to (45) and to the polynomial equivalent of the system model.

3. NON-LINEAR PREDICTIVE CONTROL – AN OVERVIEW

The development of predictive control algorithms for non-linear systems has started relatively recently. The techniques that are being used are often a direct extension of techniques for linear systems. This happens even if specific non-linear modeling methods such as neural nets or fuzzy logic are applied to prediction of future outputs. It is therefore inevitable that some sort of linearization of the system model must be performed to allow for introduction of Linear – Quadratic theory. Therefore, many researchers concentrate on defining specific conditions, which will allow linearized algorithms to work on non-linear and constrained systems. This includes the issues of stabilizability and feasibility. Another important issue is construction of efficient prediction algorithms, which would enable for fast calculation of the optimum of the performance index. A sample of different methods of non-linear predictive control is presented below.

3.1. Contractive predictive control

In this method of non-linear predictive control [28] two sampling intervals are considered. The normal sampling interval ΔT determines the frequency of changes of the control action. The "contractive" sampling interval, which is multiplicity of the normal sampling interval: $P \times \Delta T$ determines frequency of application of the contractive constraint. The system is described by a non-linear differential equation:

$$\frac{dx(t)}{dt} = f(x(t), u(t)) \qquad (47)$$

and the control action is assumed piecewise constant between sampling moments. The performance index is a quadratic form with respect to the state, the control action and the increment in the control action:

$$J(t_k) = \int_{t_k}^{t_{k+1}} x^T(t)Qx(t)dt + \sum_{i=t_k}^{t_{k+1}} u^T(i)Ru(i) + \sum_{i=t_k}^{t_{k+1}} \Delta u^T(i)\Lambda_u \Delta u(i) \qquad (48)$$

where t_k and t_{k+1} denote two sampling moments distanced by the "contractive" sampling interval. Therefore, the summations in equation (48) are performed over P steps. The predictive control law is designed to minimize the above performance index subject to the state equation constraint (equation (47)), the upper and lower constraints on the control action:

$$u_{min} \le u(i) \le u_{max}, \qquad (49)$$

the constraints on the speed of changes of the control action:

$$|\Delta u(i)| \le \Delta u_{max}, \qquad (50)$$

the constraint on the horizon of the control action:

$$\Delta u(i) = 0 \quad for \quad i = N_u, N_u+1, \ldots, P-1 \qquad (51)$$

where N_u is the control horizon,
and finally, the "contractive" constraint:

$$x^T(t_{k+1})\tilde{P}_x x(t_{k+1}) \le \alpha x^T(t_k)\tilde{P}_x x(t_k) \qquad (52)$$

where $0 \le \alpha < 1$ and $\tilde{P}_x$ is a positive definite matrix.

The control actions within the horizon $[t_k, t_{k+1})$ are calculated using a non-linear numerical optimization algorithm and then applied to the system in P sampling periods. The procedure is then repeated at the time instant t_{k+1}.

3.2. Efficient non-linear predictive control

This approach was originally developed for linear systems with constraints [26]. Several extensions have been proposed to handle non-linear systems [4, 5, 27].

The underlying idea is to split the infinite control horizon $[u(t), u(t+1), \ldots]$ into two parts. The first part $[u(t), \ldots, u(t+M)]$ will be subject to system constraints and non-linear optimization may be needed to calculate it. The second part $[u(t+M+1), \ldots]$ will be assumed linear function of the system state:

$$u(t+M+j) = Kx(t+M+j) \qquad (53)$$

Similarly, for the remote future, i.e. $t>M$, the system model will be assumed to be approximated well enough by the linearized model around zero (steady state) system state:

$$x(t+1) = Ax(t) + Bu(t) \tag{54}$$

At the end of the first part of optimization, it is assumed that the system state will fall into a so-called invariant set that guarantees feasibility and stability thereafter. It means that a linear, stabilizing control law K will maintain the system state within the set and the set itself will be assumed ellipsoidal:

$$\Omega_x = \left\{ x : x^T \Psi_x^{-1} x \le 1 \right\} \text{ where } \Psi_x \ge 0 \tag{55}$$

A semi-definite programming solver can be used to determine the maximum possible value of Ψ_x which will provide the invariance and will satisfy the input constraints:

$$|u(t)| = |Kx(t)| \le u_{max}$$

For the first *M+1* moves of the control signal, it is recognized that more freedom in the control action will be needed to compensate for constraints and non-linearities. Therefore, the control law is assumed in the form:

$$u(t) = Kx(t) + c(t) \tag{56}$$

where *c(t)* represents a correction or a "trim" to the control action and is used for optimization purposes. The optimal, predictive control problem can now be formulated as follows:
For a given non-linear system model:

$$x(t+1) = f(x(t), u(t))$$

find a set of *M* values of *c(t)* (equation (56)) such that the quadratic performance index:

$$J = \sum_{j=0}^{\infty} \left(x_{t+j+1}^T Q x_{t+j+1} + u_{t+j}^T R u_{t+j} \right) = \\ = \sum_{j=0}^{M-1} \left(x_{t+j+1}^T Q x_{t+j+1} + u_{t+j}^T R u_{t+j} \right) + x_M^T P_M x_M \tag{57}$$

is minimized and the state x_M lies in an invariant set.
The performance index (57) can be optimized numerically and, in some cases, this can be reduced to solving a linear programming task.

3.3. Fuzzy Takagi-Sugeno models in predictive control

Takagi and Sugeno proposed, [51], a type of fuzzy models suitable for the approximation of a large class of non-linear systems. This model is expressed by a set of rules R_i in the form:

$$R_i : \text{If } x_1 \text{ is } A_{i1} \text{ and} \ldots \text{and } x_n \text{ is } A_{in} \text{ then } y_i = f_i(x_1, \ldots, x_n) \tag{58}$$

where $x_1, \ldots, x_n$ are the input variables of the model; $A_{i1}, \ldots, A_{in}$ are the fuzzy sets associated to the input variables, y_i is the output of the rule *i*, and f_i is a function that could be linear, so that:

$$\mathrm{y_i} = p_0^i + p_1^i x_1 + \ldots + p_n^i \mathrm{x_n} \tag{59}$$

where p_0^i, ..., p_n^i are the consequent parameters of the rule *i*.
Combining the values of the outputs generated by all the rules in the model, the output of the fuzzy model is given by:

$$y = \frac{\sum_{i=1}^{M} \omega_i \, y_i}{\sum_{i=1}^{M} \omega_i} \tag{60}$$

with *M* as the number of the rules of the fuzzy model. Also, ω_i corresponds to the degree of satisfaction of the rule *i*, defined as:

$$\omega_i = oper\left(\mu_{A_{i1}}, \ldots, \mu_{A_{ij}}, \ldots, \mu_{A_{in}} \right) \tag{61}$$

where "*oper*" is a triangular norm given by the minimum or product operator, and $\mu_{A_{ij}}$ is the membership degree of the input variable x_j associated with the fuzzy set A_{ij}; for $j = 1, \ldots, n$.
Takagi - Sugeno dynamic models can be used to state space description of the system:

$$R_i : \text{If } x_1(t) \text{ is } A_{i1} \text{ and } \ldots \text{ and } x_n(t) \text{ is } A_{in} \\ \text{then } x^i(t+1) = A^i x(t) + B^i u(t) + C^i \tag{62}$$

where $x = [x_1, \ldots, x_n]^T$ is the state vector of the model; A^i, B^i and C^i are the matrices of the linear models in state variables for the consequences and x^i is the output vector in state variables for the rule *i*.

Presented below is a fuzzy predictive controller based on linearization of the Takagi-Sugeno fuzzy model [50]. For the fuzzy model given by equation (62), the equivalent time-variant linear model can be constructed:

$$x(t+1) = A(t)x(t) + B(t)u(t) + C(t) \tag{63}$$

where $A(t)=\sum_{i=1}^{M}\omega_i(t)A^i$, $B(t)=\sum_{i=1}^{M}\omega_i(t)B^i$ and $C(t)=\sum_{i=1}^{M}\omega_i(t)C^i$ with ω_i being the normalized satisfaction degree of the rule *i*.

At every sampling time, a linear model is derived by evaluating the fuzzy model premises or the satisfaction degrees. Then, a linear predictive controller is designed for the resulting linear model and, in the next sampling time, the linear model is updated.

4. NON-LINEAR PREDICTIVE CONTROL WITH STATE DEPENDENT STATE-SPACE MODELS

This approach is based on performing a dynamic linearization around the state trajectory and applying a receding horizon strategy. There are some similarities to the non-linear algorithms presented earlier.

Only a deterministic case is considered in this part of the paper. Extension to stochastic systems would require a careful consideration of effects which non-linearities will have on probability distributions.

4.1. System representation

Assume a non-linear, discrete in the time system described by the following equations:

State:

$$x_1(t+1)=f_1(x_1(t))+f_2(x_1(t))\cdot u(t) \qquad (64)$$

Inferred output:

$$y_h(t)=f_3(x_1(t)) \qquad (65)$$

where: f_1 is a vector of size n_x, f_2 is a matrix $(n_x\times n_u)$, f_3 is a vector of size n_y.

The above system can be transformed into an alternative representation, with state-dependent state transition matrices:

$$\begin{aligned} x_1(t+1)&=A_1(x_1(t))x_1(t)+B_1(x_1(t))u(t)\\ y_h(t)&=H(x_1(t))x_1(t)\end{aligned} \qquad (66)$$

Where: A_1 is a matrix $(n_x\times n_x)$, $B_1=f_2$, H is a matrix $(n_y\times n_x)$.

Notice that the representation (66) is not unique [9], [35], in fact there is infinite number of possible transformations leading from equation (64) to (66). For instance, if f_1 is expressed as:

$$f_1(x(t))=\begin{bmatrix} f_{11}\\ \vdots\\ f_{1n_x}\end{bmatrix}=\begin{bmatrix} f_{11}\left(x_1(t),\ldots,x_{n_x}(t)\right)\\ \vdots\\ f_{1n_x}\left(x_1(t),\ldots,x_{n_x}(t)\right)\end{bmatrix} \qquad (67)$$

then, the matrix $A_1(x_1(t))$ can be formed as follows:

$$A_1(x_1(t))=\mathrm{diag}\left\{\begin{bmatrix}\frac{f_{11}}{x_1} & \cdots & \frac{f_{1n_x}}{x_{n_x}}\end{bmatrix}\right\} \qquad (68)$$

However, also other representations are possible and, indeed, may be more appropriate, especially when the state is close to zero.

We will denote: $a_{(k)} \triangleq a(x(k))$

Using this notation and the system model as in equation (66), the *N* steps ahead prediction of the state is given by:

$$\begin{aligned} x(t+N)=&\left[A_{1(t+N-1)}A_{1(t+N-2)}\cdots A_{1(t)}\right]x(t)+\\ &+\left[A_{1(t+N-1)}A_{1(t+N-2)}\cdots A_{1(t+1)}\right]B_{1(t)}u(t)+\\ &+\left[A_{1(t+N-1)}A_{1(t+N-2)}\cdots A_{1(t+2)}\right]B_{1(t+1)}u(t+1)+\\ &+\ldots+\\ &+A_{1(t+N-1)}B_{1(t+N-2)}u(t+N-2)+\\ &+B_{1(t+N-1)}u(t+N-1)\end{aligned} \qquad (69)$$

We introduce the following notation:

$$\left[\prod_{k=l}^{n}a_{(k)}\right]\equiv\begin{cases}a_{(n)}a_{(n-1)}\cdots a_{(l)} & \text{if } l\le n\\ I & \text{if } l>n\end{cases} \qquad (70)$$

then equation (69) can be expressed as:

$$\begin{aligned} x(t+N)=&\\ &\left[\prod_{j=0}^{N-1}A_{1(t+j)}\right]x(t)+\\ &+\left[\prod_{j=1}^{N-1}A_{1(t+j)}\right]B_{1(t)}u(t)+\\ &+\left[\prod_{j=2}^{N-1}A_{1(t+j)}\right]B_{1(t+1)}u(t+1)+\\ &+\ldots+\\ &+\left[\prod_{j=N-1}^{N-1}A_{1(t+j)}\right]B_{1(t+N-2)}u(t+N-2)+\\ &+\left[\prod_{j=N}^{N-1}A_{1(t+j)}\right]B_{1(t+N-1)}u(t+N-1)\end{aligned} \qquad (71)$$

Define the following vectors, consisting of vector variables x_1, u, y_h:

$$\chi_{t,N}=\begin{bmatrix} x_1(t) \\ x_1(t+1) \\ \vdots \\ x_1(t+N-1) \end{bmatrix},\ U_{t,N}=\begin{bmatrix} u(t) \\ u(t+1) \\ \vdots \\ u(t+N-1) \end{bmatrix},$$

$$Y^h_{t,N}=\begin{bmatrix} y_h(t) \\ y_h(t+1) \\ \vdots \\ y_h(t+N-1) \end{bmatrix} \qquad (72)$$

Then,

$$\chi_{t+1,N}=\begin{bmatrix} \left[\prod_{j=0}^{0} A_{1(t+j)}\right] \\ \left[\prod_{j=0}^{1} A_{1(t+j)}\right] \\ \vdots \\ \left[\prod_{j=0}^{N-1} A_{1(t+j)}\right] \end{bmatrix} x(t)+$$

$$+\begin{bmatrix} \left[\prod_{j=1}^{0} A_{1(t+j)}\right]B_{1(t)} & 0 & \cdots & 0 \\ \left[\prod_{j=1}^{1} A_{1(t+j)}\right]B_{1(t)} & \left[\prod_{j=2}^{1} A_{1(t+j)}\right]B_{1(t+1)} & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ \left[\prod_{j=1}^{N-1} A_{1(t+j)}\right]B_{1(t)} & \left[\prod_{j=2}^{N-1} A_{1(t+j)}\right]B_{1(t+1)} & \cdots & \left[\prod_{j=N}^{N-1} A_{1(t+j)}\right]B_{1(t+N)} \end{bmatrix} U_{t,N}$$

and:

$$Y^h_{t+1,N}=\begin{bmatrix} H_{(t+1)} & 0 & \cdots & 0 \\ 0 & H_{(t+2)} & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & H_{(t+N)} \end{bmatrix} \chi_{t+1,N} \qquad (73)$$

Therefore:

$$Y^h_{t,N}=H_{N(t,t+N)}x_1(t)+G_{N(t,t+N)}U_{t,N} \qquad (74)$$

Also, the state equation (66) can be re-written in a form that contains the vector $U_{t,N}$ instead of $u(t)$. Direct analogy with (29) and (30) gives:

$$x_1(t+1)=A_{1(t)}x(t)+\beta_{(t)}u(t) \qquad (75)$$

4.2. State Dependent Riccati Equation technique

Before progressing further with design of predictive control algorithms we mention that our method is partially inspired by the State Dependent Riccati Equation technique [10], reported to be very successful for difficult mechanical systems [35]. In this approach, the system is described by a continuous equivalent of equation (66):

$$\frac{dx_1}{dt}=A_1(x_1)x_1(t)+B_1(x_1)u(t) \qquad (76)$$

A standard quadratic performance index:

$$J(t_0)=\int_{t_0}^{\infty} x_1^T(t)\tilde{Q}x_1(t)dt+\int_{t_0}^{\infty} u^T(t)\tilde{R}u(t) \qquad (77)$$

is optimized. To perform optimization, an assumption is made that for a given time instant t_0, the all future values of the system parameters are to be equal to those at time t_0 (one step ahead horizon for the system parameters changes). Consequently, the algebraic state-dependent Riccati equation (SDRE) is solved, to obtain $S(x_1)$.

$$A_1^T(x_1)S+SA_1(x_1)-SB_1(x_1)\tilde{R}^{-1}B_1^T(x_1)S+\tilde{Q}=0 \qquad (78)$$

Accepting only $S(x_1)=S^T(x_1)\geq 0 \ \ \forall \ \ x_1$.

Then, the optimal control is given by a well-known equation:

$$u(t)=-\tilde{R}^{-1}B_1^T(x)S(x)x(t) \qquad (79)$$

If equation (78) could be solved analytically it would produce an analytical equation for u. However, in the normal circumstances, it is solved numerically for a given value of x_1. Next, the system state is updated and, consequently, new system parameters are obtained. This completes one iteration of the procedure.

Theorems concerning stability of SDRE are presented in [9] and [35]. In summary, if the pair $\{A_1(x_1(t)), B_1(x_1(t))\}$ is pointwise stabilizable and the pair $\{H(x_1(t)), A_1(x_1(t))\}$ is pointwise detectable in the linear sense for all x_1 in the neighborhood of the origin, then the system controlled by the LQ regulator (79) is locally asymptotically stable.

Example of application to the flexible manipulator

The flexible manipulator is a beam pinned at the joint hinge axis, and free at the other end. The model obtained on basis of Lagrange's equations of motion yields two sets of equations. The first set is associated with the rigid body degree of freedom defined by θ, and the other set is associated with the elastic degrees of freedom defined by δ_i. These two sets of equations are nonlinear time varying coupled, second order ordinary differential equations. The system can be represented in the state-space form with six states: $x=\left[\theta_1,\theta_2,\delta_1,\delta_2,\delta_3,\delta_4\right]$ and the tip position as the output:

$$\dot{x}=f\left(x\right)+B\left(x\right)u \tag{80}$$

$$y=\theta_1+k_{tip}\left(C_1\delta_1+C_2\delta_2\right) \tag{81}$$

where, k_{tip}, C_1, C_2 are constants, depending on the arm characteristics.

A simulated example described in this section considers that the flexible link rotates on the horizontal plane. Fig. 1 shows how the tip position changes in response to a step set-point change when the system is controlled by the SDRE method. The results compare favorably with other methods, proposed for this problem in earlier literature and are therefore highly encouraging.

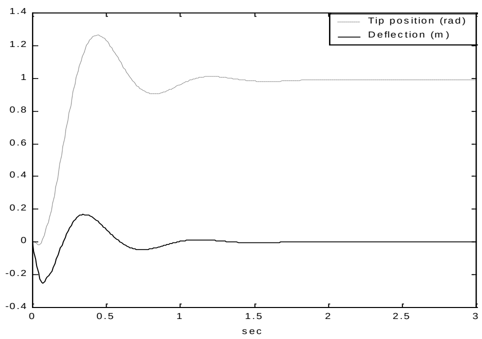

Fig. 1. Tip position, deflection for SDRE

4.3. Implementation of GPC controller

Now, we move back to the design of predictive algorithm that will use state dependent state space representation. Unlike in the previous section, we will use discrete in the time system model (66) and (74). The reference model will be introduced in the same way as for the linear system (equations (10) and (13)). Therefore, the state and the “reference state” can be combined in the extended state as in equation (14):

$$\begin{bmatrix} x_0(t+1) \\ x_1(t+1) \end{bmatrix}=\begin{bmatrix} A_0 & 0 \\ 0 & A_{1(t)} \end{bmatrix}\begin{bmatrix} x_0(t) \\ x_1(t) \end{bmatrix}+\begin{bmatrix} 0 \\ B_{1(t)} \end{bmatrix}u(t) \tag{82}$$

Notice that the lower part of this system, i.e. $A_{1(t)}$ and $B_{1(t)}$ represents the non-linear state dependent behavior. The standard predictive control performance index will cover finite number of N steps into the future and therefore can be expressed as in (22):

$$J_t=\left\{\left(R_{t+1,N}-Y^h_{t+1,N}\right)^T\tilde{Q}\left(R_{t+1,N}-Y^h_{t+1,N}\right)+U^T_{t,N}\tilde{R}U_{t,N}\right\} \tag{83}$$

The solution of the optimization problem is given by:

$$U_{t,N}=\left(G^T_{N(t,t+N)}\tilde{Q}G_{N(t,t+N)}+\tilde{R}\right)^{-1}\times G^T_{N(t,t+N)}\tilde{Q}\left(R_{t,N}-H_{N(t,t+N)}x_1(t)\right) \tag{84}$$

The above control law is optimal for the non-linear system (82) and the quadratic performance index (83). However, the control law (84) is not causal. The matrices in the solution are functions of the future system states, which are not known when the control is calculated. Similarly to the section 4.2, a causal solution could be obtained if the analytical relationships from (82) are substituted to the prediction equation and the resulting equation is solved for *u*. Alternatively, one can calculate the control action iteratively as follows:

1. The current time instant is *t*

PART A: Initial conditions for the current time instant

2. obtain the state *x(t)*
3. obtain the matrices *A(x(t)), B(x(t)), H(x(t))*
4. assume that those matrices will remain constant for the next *N* steps
5. based on this (linear model) calculate the state predictions, the output predictions and the control vector $U_{t,N}$

PART B: Iterations performed within one time instant

6. substitute the calculated $U_{t,N}$ into the state equation (66) and calculate iteratively the state predictions and associated state matrices:
   $x(t+1)\rightarrow A_{(t+1)}, B_{(t+1)}, H_{(t+1)}$
   …
   $x(t+N-1)\rightarrow A_{(t+N-1)}, B_{(t+N-1)}, H_{(t+N-1)}$
7. Calculate the output predictions and the control vector $U_{t,N}$
8. check the difference between the state predictions now and in the previous iteration step, and between the control vector $U_{t,N}$ calculated now and in the previous iteration step
9. If the difference is not small enough: go back to 6.
   If the difference is small enough: end iterations (Part B)
10. Increase current time index by 1 and go to 2.

4.4. Implementation of LQGPC controller

Following the reasoning presented in section 2.4 the LQGPC (dynamic performance) index will be formulated as a sum of the indices defined by (83), i.e.:

$$J=\frac{1}{T+1}\sum_{j=0}^{T}\left\{\left(R_{t+j+1,N}-Y^h_{t+j+1,N}\right)^T\tilde{Q}\left(R_{t+j+1,N}-Y^h_{t+j+1,N}\right)+U^T_{t+j,N}\tilde{R}U_{t+j,N}\right\} \tag{85}$$

Using the state as defined in equation (75) and the input and output as defined in equation (74) the problem can be solved in exactly the same way as in section 2.4, leading to coupled (algebraic) Riccati equations as in (37) and (38).

Notice that in this formulation, knowledge of N future states is required to construct the state-space model. Therefore, as before, the obtained solution is not causal. However, it can be approximated by an iterative procedure similar to the one described in the section above.
As before, it is assumed that at the time instant t_0 it is possible to predict N future values of control signals and therefore N future values of states of the system. Furthermore, it is assumed that the plant parameters beyond this horizon will be constant. This assumption has no implication when using performance index (83) as the optimization is performed in one step (static optimization). However, if the performance index (85) is used, the optimization problem within the horizon T+1 will require a solution of Riccati equation backward from T+1 to 1. In the iterations of the Riccati equation the last N steps (i.e. the first N steps in time) will feature changing parameters of the system.

5. EXAMPLE

Two examples are presented in this section. The first, which illustrates the state dependent GPC technique, is based on a simplified model of a helicopter. The model is a multidimensional naturally unstable system with two manipulated inputs and two measured outputs with significant cross-couplings. The model is described by non-linear state-space equations with two inputs, two outputs and nine states. The inputs are: the throttle valve opening for the main propeller and for the side propeller and the outputs are: elevation angle and azimuth angle.

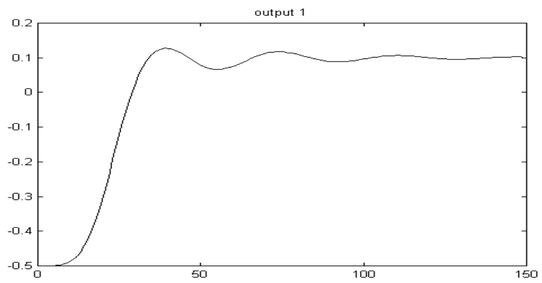

Fig. 2. Elevation angle with non-linear GPC

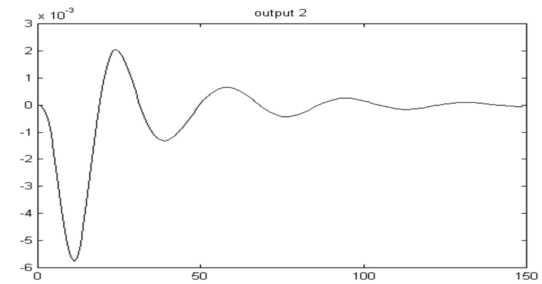

Fig. 3. Azimuth angle with non-linear GPC

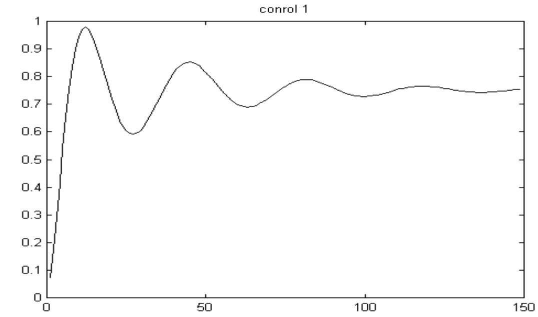

Fig. 4. Main propeller motor control

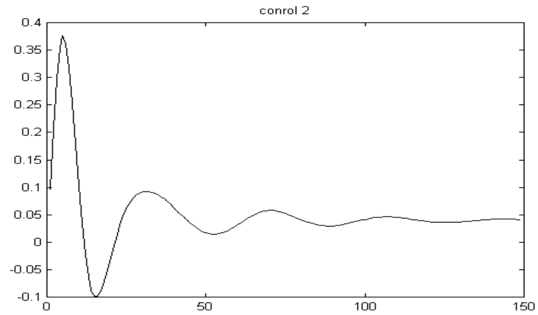

Fig. 5. Side propeller motor control

For elevation angles greater than 90 degrees helicopter model is unstable. For this model, two different control algorithms have been tried. The first was based on linear GPC technique, i.e. the system non-linear equations were linearized in the current operating point and then the linear GPC solution computed and the first control applied. In the next time instant the linearization and the computations of GPC controller were repeated. This is very similar to the SDRE technique but with a finite horizon. Using this approach, all the attempts to stabilize the helicopter model failed. Then, the approach described in section 4.3 was tried and the system was successfully stabilized with the GPC non-linear controller. The results are presented in Fig. 2 to Fig. 5.

The second example refers to the flexible manipulator described earlier. Fig. 6 shows the results of application of the state-dependent GPC controller and Fig. 7 refers to the non-linear LQGPC control, both described in the previous paragraphs. Both techniques compare favorably with the SDRE (see Fig. 1), and LQGPC is slightly better than GPC. It is worth mentioning that the SDRE technique already provides very good results to this difficult problem. The algorithms based on predictive approach prove easier to tune which can be credited to a larger number and greater transparency of tuning parameters.

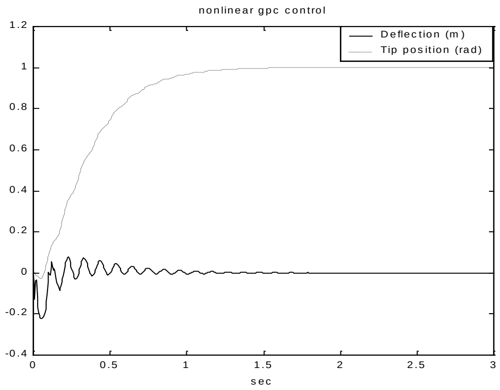

Fig. 6. Tip position, deflection for GPC control

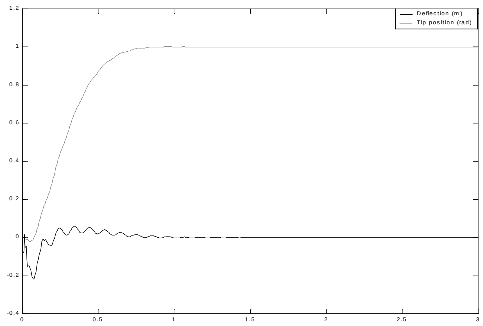

Fig. 7. Tip position, deflection for nonlinear LQGPC control.

6. CONCLUSIONS

I a recent excellent tutorial session [48], Jacques Richalet, whose contribution to the field of predictive control, e.g. [49,47] is unanimously appreciated, specified 4 basic principles by which all those algorithms which pretend to be named "predictive" should be judged. Those principles are:

- internal model; i.e. the model of the system is known and used internally in the control algorithm to predict the future outputs
- reference trajectory, is defined for a finite number of steps into the future
- structurization of manipulated variables, i.e. the control action is approximated by a combination of pre-defined functions of time
- self-compensation.

However, one may find out that those principles are not fully obeyed in the majority of predictive control literature. Many researchers with a background in LQG optimization wish to see the predictive control as a special case (namely finite, receding horizon) of LQG problem. Whereas the four principles listed above do not even mention optimization, only a definition of a reference trajectory.

Also, we admit that this presentation is biased by the "Linear-Quadratic optimization" thinking. However, trying to bear in mind that the Model Based Predictive Control is a more general approach, not a special case of the LQG method, let us consider some consequences this may have on design of algorithms, especially in non-linear and/or constrained cases.

The reference trajectory:
Very often the papers on predictive control contain the sentence: *without loss of generality assume zero reference signal*. This is not necessarily so easy, especially in the real applications where the human operator or the higher level in the automatic control hierarchy would need to have a facility to change the value of the reference signal on-line while the algorithm is running. We believe that the way of describing the reference signal provided in section 2.1 addresses this problem.

Set-point and stability:
Many publications base the stability analysis on methods that assume the equilibrium point at the origin in the state space [25, 29]. For non-zero reference signal the state of the system not necessarily stabilizes at zero and in non-linear and constrained cases this will strongly affect the stability and feasibility considerations.

Structurization of manipulated variables:
This approach is now gaining popularity due to its promising features and computational advantages for non-linear systems [14, 31, 46]. Notice that standard, LQG based predictive control algorithms can provide zero steady state error for constant reference signal. For reference being a ramp function the tracking error will be constant, determined by the closed loop gain. However, when using the appropriately structured manipulated variables it is possible [47] to achieve zero steady-state error for a ramp reference function or even faster changing reference signals (e.g. quadratic function of time).

Reduced order controller:
As a consequence of structurization of manipulated variables, the order of the controller can be pre-specified at the design stage and can be lower than the dimension of the system state. The price paid for this is that the parameters of the controller will have to change from step to step [55]. However, in most of non-linear predictive algorithms even without structurization of the manipulated variables, the controller is being re-calculated in every step and still results in a high order dynamic system. Being able to reduce the order of the controller dynamics may be an attractive feature if a computational power is limited or a particular hardware imposes restrictions on the controller structure.

Acknowledgments

The authors wish to thank Mr. Alaa Eldien Shawky Mohamedy and Mr. Arkadiusz Dutka for providing the application examples and running the simulations. Comments from Mr. Baris Bulut are acknowledged.
This work was partially supported by the Engineering and Physical Science Research Council (EPSRC)

under grant *Non linear predictive control and industrial applications*.
Also, a support from the European Commission grant *European Network on Polynomial Design Methods* is acknowledged.

REFERENCES

1. Bitmead R.R., M. Gevers and V. Wertz, *Adaptive Optimal Control. The Thinking Man's GPC*. Prentice Hall, 1990,
2. Camacho E.F. and C. Bordons, *Model Predictive Control*, Springer Verlag, 1999,
3. Camacho, E. (1993). *Constrained generalized predictive control*, IEEE Transactions on Automatic Control, Vol. 38, Nº 2, pp. 327-332,
4. Cannon M. and B. Kouvaritakis, *Continuous time predictive control of constrained non-linear systems*, In: Progress in Systems and Control Theory, Vol 26, Birkhausen, Verlag Basel/ Switzerland, 2000,
5. Cannon M., B. Kouvaritakis, Y. Lee and A.C. Brooms, Efficient non-linear model based predictive control, International Journal of Control, vol. 74, No 4, pp. 361-372, 2001,
6. Chisci L., A. Lombardi and E. Mosca, *Dual receding horizon control of constrained discrete-time systems*, European Journal of Control, Vol. 2, pp. 278-285, 1996,
7. Clarke, D.W., C. Mohtadi and P.S. Tuffs, 1987, *Generalised Predictive Control - Part 1. The basic algorithm*, Automatica, Vol. 23, No. 2, pp.137-148,
8. Clarke, D.W., C. Mohtadi and P.S. Tuffs, 1987, *Generalised Predictive Control - Part 2. Extensions and Interpretations*, Automatica, Vol. 23, No. 2, pp.149-160,
9. Clarke D.W. and R. Scattolini, *Constrained receding horizon predictive control*, Proc. IEE, Part D, 138, pp.347-354, 1991,
10. Cloutier J.R., State-Dependent Riccati Equation Techniques: An Overview, Proc. ACC, Albuquerque, New Mexico, June 1997
11. Farinwata S.S., D. Filev and R. Langari (Editors), *Fuzzy Control, synthesis and analysis*, Wiley, Chichester, 2000,
12. Gambier A. and H. Unbehauen, A state-space Generalized Model-Based Predictive Control for linear multivariable systems and its interrelation with the receding horizon LQG optimal control, Proc. CDC, San Antonio, 1993,
13. Garcia C.E., D.M. Prett and M. Morari, *Model Predictive Control: Theory and Practice-A Survey*, Automatica, 25(3), 1989,
14. Gawthrop P., Physical-Model-Based Predictive Control, in: ACT Club meeting on Non-linear Predictive Control, Oxford, Nov. 2000,
15. Grimble, M.J., *Robust Industrial Control*, Prentice Hall, Hemel Hempstead, 1993,
16. Grimble, M.J. and Johnson, M.A. *Optimal control and stochastic estimation theory*, Parts I and II. Wiley, 1988,
17. Grimble, M.J., 1995, *Two DOF LQG predictive control*, IEE Proc. Vol. 142, No. 4, July, pp. 295-306,
18. Grimble, M.J., 1998, *Multi step H_∞ generalized predictive control*, Dynamics and Control, 8, pp. 303-339,
19. Grimble, M.J., 1979, *Optimal control of linear systems with crossproduct weighting*, Proc. IEE, Vol. 126, No. 1, pp. 95-103,
20. Grimble, M.J., 1979, *Design of optimal stochastic regulating systems including integral actions*, Proc. IEE, Vol. 126, No. 9, pp. 841-848
21. Grimble M.J. and V. Kucera (editors), *Polynomial Methods for Control System Design*, Springer Verlag, 1996,
22. Grimble M. and Ordys A.W., *Predictive Control for Industrial Applications*, Plenary Lecture at IFAC Control System Design Conference, Bratislava, 18-20 June, 2000,
23. Grimble M.J. and Ordys A.W., Predictive control design for systems with state dependent non-linearities, SIAM Conference on Control and Its Applications, San Diego, July 2001,
24. Kerrigan E.C., Robust Constraint Satisfaction: Invariant Sets and Predictive Control, Doctor of Philosophy Dissertation, University of Cambridge, Department of Engineering, 2000,
25. Kokotovic P. and Arcak M., Constructive nonlinear control: a historical perspective, Automatica, 37, pp 637-662, 2001,
26. Kouvaritakis B., J.A. Rossitier and J. Schuurmans, Efficient robust predictive control, IEEE Transactions on Automatic Control, vol. 45, No.6, June 2000,
27. Kouvaritakis B., J. A. Rossitier and M. Cannon, *Linear quadratic feasible predictive control*, Automatica, vol 34, No 2, pp. 1583-1592, 1998,
28. Loureiro de Oliveira Kothare S. and M. Morari, Contractive Model Predictive Control for Constrained Nonlinear Systems, IEEE Transactions on Automatic Control, vol. 45, No. 6, June 2000
29. Mayne D.Q., J.B. Rawlings, C.V. Rao and P.O.M. Scokaert, Constrained model predictive control: Stability and optimality, Automatica, 36, pp. 789-814, 2000
30. Martin-Sanchez J.M. and J. Rodellar, *Adaptive Predictive Control. From the Concepts to Plant Optimization*, Prentice Hall, 1996,
31. Morningred J. Duane, D. a. Mellichamp and D. Seborg, *A Multivariable Adaptive Nonlinear Predictive Controller*, ACC, Boston,

Massachesetts, June 26-28, 1991, Vol. Pp 2776-2781,

32. Morrari M. and E. Zafiriou, *Robust Process Control*, Prentice Hall, 1989,
33. Mosca E., *Optimal Predictive and Adaptive Control*, Prentice Hall, 1995,
34. Mosca E. and J. Zhang, *Stable redesign of predictive control*, Automatica, Vol. 28 No 6, pp. 1129-1233, 1992,
35. Mracek C.P. and Cloutier J.R., Control Designs for the nonlinear benchmark problem via the state-dependent Riccati equation method, Int. J. of Robust and Nonlinear Control, 8, pp. 401-433, 1998,
36. Ordys A.W., J. Stoustrup and J. Smillie, *Comparison of H_∞ and Generalized Predictive Control for a laser scanner system*, Proceedings of IMechE, Journal of Systems and Control Engineering, Vol 214 Part I, 2000, pp. 283-297,
37. Ordys A.W. , M. Hangstrup and M.J. Grimble *Dynamic algorithm of LQGPC Predictive Control*, Int. Journal of Applied Mathematics and Computer Science, vol.10, No.2,2000, pp. 101-118,
38. Ordys A.W. and D.W. Clarke, *A state-space description for GPC controllers*, International Journal of Systems Science, Vol.24, No.9, 1993,
39. Ordys A.W. and M. J. Grimble, *Evaluation of stochastic characteristics for a constrained GPC algorithm*, in : Advances in Model Based Predictive Control, editor : David Clarke, Oxford University Press, 1994,
40. Ordys A.W. and M.J. Grimble, *A multivariable dynamic performance predictive control with application to power generation plants*, IFAC World Congress, San Francisco July, 1996,
41. Ordys A. W. and M.J. Grimble, *Predictive Control in Power Generation*, IEE Computing and Control Engineering Journal, vol.10, No.5, October 1999,
42. Ordys A.W. and P. Kock, *Constrained Predictive Control for Multivariable Systems with Application to Power Systems*, International Journal of Robust and Non-Linear Control, August, 1999,
43. Ordys A.W., *Steady-state offset in predictive control*, American Control Conference, San Diego, USA, June 1999,
44. Ordys A.W. and A.W. Pike, *State-space Generalized Predictive Control incorporating direct through terms*, 37th IEEE Control and Decision Conference, Tampa, Florida, Dec. 1998,
45. Pike A.W., M.J. Grimble, S.Shakhour and A.W. Ordys, *Predictive Control*, in The Control Handbook, editor W.S. Levine, CRC Press INC, 1995,
46. Pottmann Martin and Dale E. Seborg, *Nonlinear Predictive Control Strategy Based on Radial Basis Function Models*, Computers Chemical Engineering, Vol. 21, No.9, pp 965-980, 1997,
47. Richalet J., *Pratique de la commande predictive*, Hermes, Paris 1993 (in French),
48. Richalet J., Tutorial session on Model Based Predictive Control, Training course organised by EU project DYCOMANS, Budapest, March 2001,
49. Richalet, J, A, Rault, L, Testaud and J, Papon, 1978, *Model predictive heuristic control:* Applications to industrial processes, Automatica, Vol. 14. pp.413-428,
50. Saez D., A. Cipriano and A. Ordys, *Optimisation of Industrial Processes at Supervisory Level, Application to Control of Thermal Power Plants,* Springer Verlag, to appear 2001,
51. Takagi, T., Sugeno, M. (1985). *Fuzzy identification of systems and its applications to modeling and control*, IEEE Transactions on Systems, Man and Cybernetics, Vol. SMC-15, Nº1, pp. 116-132
52. Tomizuka M., *Optimum linear preview control with application to vehicle suspension – revisited.* Transactions of the ASME, Journal of Dynamic Systems, Measurement and Control, Vol.98, No.3, pp.309-315, September 1976,
53. Tomizuka M. and D.E. Rosenthal, *On the optimal digital state vector feedback controller with integral and preview actions.* Transactions of the ASME Vol.101, pp.172-178, June 1979,
54. Tomizuka M. and D.E. Whitney, *Optimal discrete finite preview problems (why and how is future information important?)*, Transactions of the ASME, Journal of Dynamic Systems, Measurement and Control, Vol.97, pp.319-325, December 1975,
55. Uduehi D., A. Ordys and M.J. Grimble, *A one degree of freedom interpretation for GPC controller using time-varying restricted order controllers*, SIAM Confrence on Control and Its Applications, San Diego, July 2001,
56. Vilanova R., A.W. Ordys and M.J. Grimble, *Linear Quadratic Gaussian Predictive Control: Analysis of Stability Properties*, Process Control and Instrumentation 2000, Glasgow, 26-28 July 2000.