Proceedings of the
2005 IEEE Conference on Control Applications
Toronto, Canada, August 28-31, 2005

Parameterized polyhedra approach for robust constrained generalized predictive control

Sorin Olaru, Didier Dumur and Sihem Tebbani

***Abstract*— The paper considers the discrete time-invariant linear systems affected by input disturbances and construct the explicit description of the constrained Generalized Predictive Control (GPC) law taking in account the constraints existence from the design stage.**

The explicit formulation of the predictive controllers gives a useful insight on the closed loop capabilities. The GPC is a special case of Model Predictive Control (MPC) whose explicit formulation is known to be a piecewise affine function of state. As novelty the present work shows that this piecewise linear dependence on the context parameters can be found by exploiting the fact that the optimum of a min-max multiparametric program is placed on the vertices of a parameterized polyhedron. As these parameterized vertices have specific validity domains, the control law is expressed as a piecewise linear function of the current system parameters.

The resulting GPC law is formulated in terms of a look-up table with two-degree of freedom polynomials in the backward shift operator (also known as the RST formulation).

I. Introduction

Predictive Control is an optimization based control design technique which enjoys a remarkable reputation for its simplicity and versatility. Its unconstrained formulation is known to result in a low complexity linear control law which can further be completed with a mechanism [6] to deal with constraints violation a posteriori. However, due to the time domain formulation of the predictive laws, the inclusion of constraints from the design stage was investigated with excellent results towards stability, feasibility or robustness [9]. The drawback is the relatively high complexity of the optimization problem to be solved at each sampling time.

Improvements in the on-line computations can be achieved if the explicit solution of the multiparametric optimization problem from the background of the predictive law is elaborated. Thus at each sampling time, a piecewise linear function has to be evaluated instead of running an iterative optimization routine. In the nominal case corresponding with a quadratic optimization problem and linear constraints, the explicit solution was investigated with success using an algebraic approach in [1], geometrical arguments [4], [11] and dynamic programming [4]. These alternatives with different maturity degrees converged towards similar formulations and represent valuable results.

In the case of robust MPC, the explicit solution is somehow more difficult to achieve as the optimization problem is based on a min-max cost function. It was successfully tackled in the studies of [2], but the alternative methods had a small delay in presenting similar solutions.

The current work is trying to compensate this setback by presenting an explicit solution for the robust predictive control through geometrical arguments in the continuity of [11]. The approach is based on the concept of parameterized polyhedra [8] and related parameterized vertices, where the optimal solution can be found. Another particularity will be the specific focus on Generalized Predictive Control (GPC) which has a great success in industry applications [3].

In the following, Section 2 formulates the robust GPC problem and the related optimization. The main result is presented in Section 3 in terms of the explicit multiparametric linear optimization problem defining the robust GPC law. In Section 4 the design procedure is applied for an application to a positioning benchmark including an induction machine.

This work was supported by the European Commission, Directorate for Research.

All authors are with the Automatic Control Department, Supelec, Plateau de Moulon, 91192 Gif-sur-Yvette, France. sorin.olaru@ieee.org; didier.dumur@supelec.fr

II. Robust Generalized Predictive Control

A. GPC formulation

Model-based Predictive Control implies the idea of minimizing a cost function based on the predicted plant evolution. This strategy is also called the "receding horizon principle" and differs from one algorithm to another by the plant model chosen or by the cost function considered. The Robust Generalized Predictive Control (RGPC) is characterized by two major features:

- It uses a CARIMA plant model:

$$A(q^{-1})\, y_t = B(q^{-1})\, u_{t-1} + \underbrace{\xi_t/\Delta(q^{-1})}_{v_t} \qquad (1)$$

  with u, y the system input and output, ξ_t a centered Gaussian white noise, A and B polynomials in the backward shift operator q^{-1} of respective degree n_a and n_b, and $\Delta = 1 - q^{-1}$ the difference operator.
- The cost function is penalizing the tracking error and control effort over a receding horizon:

$$U = \sum_{j=N_1}^{N_2} |w_{t+j} - \hat{y}_{t+j}| + \sum_{j=1}^{N_u} \lambda_j \, |\Delta u_{t+j-1}| \qquad (2)$$

  with $\hat{y}(t+j)$ the output prediction, N_1, N_2 the prediction horizons, N_u the control horizon, λ_j the control weighting factor and w the set-point. Note that the cost function U corresponds to the sum of ∞ -norm terms in other robust MPC formulations.

Using this model, the j-step ahead predictor will be:

$$\hat{y}_{t+j} = \underbrace{F_j(q^{-1})y_t + H_j(q^{-1})\Delta u_{t-1}}_{\text{freeresponse}} + \\ + \underbrace{G_j(q^{-1})\Delta u_{t+j-1} + \tilde{J}_j(q^{-1})v_{t+j}}_{\text{forced response}} \quad (3)$$

where the F_j, G_j, H_j, J_j polynomials are solution of the Diophantine equations:

$$\begin{array}{l} \Delta(q^{-1})A(q^{-1})J_j(q^{-1}) + q^{-j}F_j(q^{-1}) = 1 \\ G_j(q^{-1}) + q^{-j}H_j(q^{-1}) = B(q^{-1})J_j(q^{-1}) \end{array} \quad (4)$$

and $\tilde{J}_j(q^{-1}) = J_j(q^{-1})\Delta(q^{-1})$ is multiplying the disturbance v_{t+j}. The disturbances are included in a polyhedral domain containing the origin:

$$V = \{v \,| Mv \leqslant l; l \geqslant 0\} \quad (5)$$

Remark: The coefficients of G_j represent in fact the step response coefficients for the system described by the polynomials $(A(q^{-1}), B(q^{-1}))$. Similarly, J_j are formed with the impulse response coefficients of the system described by the polynomials $(A(q^{-1}), 1)$.

B. Robust GPC

Robust GPC has to consider the worst combination of disturbances and thus to solve at each sampling time a min-max optimization problem, then only the first component of the optimal sequence $\mathbf{k}_u$ is effectively applied, as at the next sampling time "a receding horizon strategy" is followed:

$$\min_{\mathbf{k}_u(t)} \max_{\mathbf{k}_v(t)} \sum_{j=N_1}^{N_2} |\hat{y}_{t+j} - w_{t+j}| + \sum_{j=1}^{N_u} \lambda_j \,|\Delta u_{t+j-1}| \quad (6)$$

with the vectors $\mathbf{k_u} = \left\{\Delta u_t, \ldots, \Delta u_{t+N_u-1}\right\}$ and $\mathbf{k}_v = \{v_{t+1}, \ldots, v_{t+N_2}\}$. The optimization (6) has to be solved subject to diverse types of constraints rising from operational or performance considerations. These are represented generally as a set of linear inequalities:

$$\left\{ \begin{array}{l} C\mathbf{k}_u + D\mathbf{k}_v \leqslant d + E\mathbf{p}, \\ Mv_{t+k} \leqslant l, k = 0, \ldots N_2 - 1 \\ \hat{y}_{t+N_2} \in Y_f \end{array} \right. \quad (7)$$

where $\mathbf{p} = \left[u_{t-1} \cdots u_{t-n_b}\, y_t \cdots y_{t-n_a}\, w_{t+N_1} \cdots w_{t+N_2}\right]^T$, Y_f is a terminal set and the terminal constraint in (7) is imposed from the stability considerations.

Remark: The terminal constraints in the classical GPC formulations are represented by equality constraints at the end of the prediction horizon. In the robust GPC, such constraints are very difficult to fulfill as they cancel the control degrees of freedom implying severe feasibility problems.

The constrained optimization (6-7) provides a robust control sequence but is quite conservative as it is considered for all disturbance realizations ignoring that measurements are available as time progresses. The control potential is improved if a feedback approach is adopted resulting in a nested min-max formulation:

$$\begin{array}{l} \min\limits_{\Delta u_t} \{\lambda_0 \,|\Delta u_t| + \max\limits_{v_t} \{|\hat{y}_{t+1} - w_{t+1}| + \\ + \min\limits_{\Delta u_{t+1}} \{\ldots + \min\limits_{\Delta u_{t+N_u-1}} \{\lambda_{N_u-1} \,|\Delta u_{t+N_u-1}| + \\ + \max\limits_{v_{t+N_u-1},\ldots,v_{t+N_2-1}} \sum\limits_{k=N_u}^{N_2} |\hat{y}_{t+k} - w_{t+k}| \Bigg\}\Bigg\} \cdots\}\} \end{array} \quad (8)$$

subject to (7). This represents a "closed loop" formulation [12], avoiding the feasibility problems in comparison with the "open-loop" formulations.

The robust GPC formulation is based on the on-line solving of the associated min-max optimization problem (6-7) or (7-8). Both the cost function and the set of constraints depend on the parameter's vector $\mathbf{p}$. The alternative solution to the on-line iterative routines is to explicitly formulate off-line the optimal solution $\mathbf{k}_u^*(\mathbf{p})$ further evaluate it on-line.

C. The linear multiparametric optimization

The disturbances in the form (6) can be considered using the extremal possible combinations of vertices in V for each prediction stage completing the sequence $\mathbf{k}_v$:

$$v_t \in V \subset \mathbb{R} \Rightarrow \mathbf{k}_v \in V^{N_2} \subset \mathbb{R}^{N_2} \quad (9)$$

The optimum for the inner optimization in (6) will be on the border of the feasible domain, more precisely on one of the vertices of V^{N_2} as long as it is defined as a polytope and the objective function is convex. Thus (6) becomes:

$$\begin{array}{l} \min\limits_{\mathbf{k_u}} \max\limits_{\mathbf{v}_{\mathbf{k}_{v_l}}} U(\mathbf{p}, \mathbf{k_u}, \mathbf{k}_{v_l}) \\ \text{subj.to} : A_{in_1}\mathbf{k_u} + A_{in_2}\mathbf{k}_{v_l} \leqslant b_{in} + B_{in}\mathbf{p} \\ l \in L, \mathbf{k}_{v_l} \in V^{N_2} \end{array} \quad (10)$$

with $L = \{1, 2, \ldots, N_v\}$ and $N_v = 2^{N_2}$ is the number of vertices in V^{N_2}. This means that the inner optimization in (6) will act only on the set of vertices of V^{N_2}. The inequalities in (10) represent a compact rewritten of those in (7).

The main impediment in finding the explicit formulation of the solution for (10) is the expression of the cost function, given as a collection of ∞-norm terms. In order to avoid the inherent difficulty of handling it, an equivalent linear program (LP) [5] formulation must be achieved. The optimization problem is equivalent with the minimization of the sum of bounds for these terms. This is resumed by the following result where the cost function is considered as a sum of linear terms in the vector of unknowns $\mathbf{k_u}$ and parameters $\mathbf{p}$ (the disturbance vector is ignored at this stage).

Proposition 1: Relations (a) and (b) are equivalent:

(a) $K(\mathbf{p}) = \min\limits_{\mathbf{k}_u} U(\mathbf{k}_u, \mathbf{p}) = \min\limits_{\mathbf{k}_u} \sum\limits_{i=1}^{n} \|S_i\mathbf{k}_u + P_i\mathbf{p} + s_i\|_\infty$
subject to $A_{in}\mathbf{k}_u \leqslant b_{in} + B_{in}\mathbf{p}$

(b) $K(\mathbf{p}) = \min\limits_{\rho, \{\sigma_i\}, \mathbf{k}_u} \rho$

subject to $\left\{ \begin{array}{l} -\mathbf{1}\sigma_i \leqslant S_i\mathbf{k}_u + P_i\mathbf{p} + s_i \leqslant \mathbf{1}\sigma_i, 1 \leqslant i \leqslant n \\ \sum\limits_{i=1}^{n} \sigma_i \leqslant \rho \\ A_{in}\mathbf{k}_u \leqslant b_{in} + B_{in}\mathbf{p} \end{array} \right.$

with $\sigma_i, \rho \in \mathbb{R}$, $\mathbf{1}$ a unit entries vector of appropriate length and n the number of ∞-norm terms in U.

With the previous transformations the optimization (6) can be rewritten as a compact multiparametric linear program:

$$\mathbf{k_u} * (\mathbf{p}) = \underset{\rho, \mathbf{k_u}, \{\sigma_i^j\}}{\arg\min} \ \rho$$
$$\begin{cases} -\mathbf{1}\sigma_i^j \leqslant S_i \mathbf{k_u} + P_i \mathbf{p} + W_i \mathbf{k}_{v_l} + s_i \leqslant \mathbf{1}\sigma_i^j, \\ \qquad\qquad 1 \leqslant i \leqslant n, 1 \leqslant l \leqslant N_v \\ \begin{bmatrix} \sum_{i=1}^{n} \sigma_i^1 \\ \vdots \\ \sum_{i=1}^{n} \sigma_i^{N_v} \end{bmatrix} \leqslant \mathbf{1}\rho \\ A_{in_1}\mathbf{k_u} + A_{in_2}\mathbf{k}_{v_l} \leqslant b_{in} + B_{in}\mathbf{p}, \\ \qquad\qquad 1 \leqslant l \leqslant N_v \end{cases} \tag{11}$$

III. THE EXPLICIT SOLUTION

In the following, the closed form of the RGPC law is the main objective which implies the construction of the explicit solution of multi-parametric linear programs (MPLP) (11). A procedure will be proposed focusing on the set of constraints and its geometrical correspondence [11] using the concept of parametrized polyhedron. Thus the resulting algorithm differs from the solutions based on branch and bound methods or other mixed integer linear solvers, being mainly focused on the enumeration of the edges of an augmented dimension polyhedron. In order to get familiar with these concepts, some brief definitions are introduced (details can be found in [7], [8]). It is noted in this part $\mathbf{x} = \mathbf{k}_u$ for the simplicity of development ($\mathbf{x} \in \mathbb{R}^n, \mathbf{p} \in \mathbb{R}^m$).

A. Parameterized Polyhedra

A system of linear constraints as in (11) defines a polyhedron [8]:

$$P = \{\mathbf{x} \in \mathbb{R}^n \,|\mathbf{A}_{eq}\, \mathbf{x} = \mathbf{b}_{eq}; \mathbf{A}_{in}\mathbf{x} \leqslant \mathbf{b}_{in}\} \tag{12}$$

or in a dual Minkowski representation of generators:

$$P = conv.hull\,\{\mathbf{x}_1, .., \mathbf{x}_v\} + cone\,\{\mathbf{y}_1, .., \mathbf{y}_r\} + lin.space\mathbf{Z} \tag{13}$$

where *conv.hullX* denotes the set of convex combinations of points in ***X*** , *coneY* denotes nonnegative combinations of unidirectional rays and *lin.spaceZ* represents a linear combination of bidirectional rays. It can be also rewritten:

$$\begin{aligned} &P = \left\{ \mathbf{x} \in \mathbb{R}^n \middle| \, \mathbf{x} = \sum_{i=1}^{v} \lambda_i \mathbf{x}_i + \sum_{i=1}^{r} \gamma_i \mathbf{y}_i + \sum_{i=1}^{l} \mu_i \mathbf{z}_i \right\} \\ &0 \leqslant \lambda_i \leqslant 1,\ \sum_{i=1}^{v} \lambda_i = 1\ ,\ \gamma_i \geqslant 0\ ,\ \forall \mu_i \end{aligned} \tag{14}$$

A *parameterized polyhedron* is defined in the implicit form by a finite number of inequalities and equalities with the note that the affine part depends linearly on a parameter vector $\mathbf{p}$:

$$\begin{aligned} P'(\mathbf{p}) &= \{\mathbf{x} \in \mathbb{R}^n \,|\mathbf{A}_{eq}\, \mathbf{x} = \mathbf{B}_{eq}\mathbf{p} + \mathbf{b}_{eq}; \\ &\qquad \mathbf{A}_{in}\mathbf{x} \leqslant \mathbf{B}_{in}\mathbf{p} + \mathbf{b}_{in}\} \\ &= \left\{ \mathbf{x}(\mathbf{p}) \middle| \, \mathbf{x}(\mathbf{p}) = \sum_{i=1}^{v} \lambda_i(\mathbf{p})\mathbf{x}_i(\mathbf{p}) + \right. \\ &\qquad \left. + \sum_{i=1}^{r} \gamma_i \mathbf{y}_i + \sum_{i=1}^{l} \mu_i \mathbf{z}_i \right. \\ &0 \leqslant \lambda_i(\mathbf{p}) \leqslant 1,\ \sum_{i=1}^{v} \lambda_i(\mathbf{p}) = 1\ ,\ \gamma_i \geqslant 0\ ,\ \forall \mu_i \end{aligned} \tag{15}$$

with $\mathbf{z}_i$ the lines, $\mathbf{y}_i$ the rays, $\mathbf{x}_i$ the vertices and μ_i, γ_i, λ the corresponding coefficients.

Remark: Only the vertices are concerned by the parametrization of the polyhedron (parameterized vertices - $\mathbf{x}_i(\mathbf{p})$, the rays and the lines do not change with the parameters' variation.

The parameterized polyhedron $P'(\mathbf{p})$ has a non-parameterized correspondent in an augmented space:

$$\begin{aligned} \tilde{P}' = &\left\{ \begin{pmatrix} \mathbf{x} \\ \mathbf{p} \end{pmatrix} \in \mathbb{R}^{n+m} \middle| \right. \\ &\left. \begin{bmatrix} \mathbf{A}_{eq/in} | -\mathbf{B}_{eq/in} \end{bmatrix} \begin{pmatrix} \mathbf{x} \\ \mathbf{p} \end{pmatrix} = \mathbf{b}_{eq/in} \right\} \end{aligned} \tag{16}$$

The form (15) faces an important difficulty, the description of the parameterized vertices $\mathbf{x}_i(\mathbf{p})$ and their validity domains. This can be achieved using the fact that they correspond to m-faces in the augmented data ($\mathbb{R}^n$)+parameter($\mathbb{R}^m$) space for the polyhedron (16):

$$\mathbf{x}_i(\mathbf{p}) = \mathrm{Proj}_n \left(F_i^m(\tilde{P}') \cap S(\mathbf{p}) \right) \tag{17}$$

where $\mathrm{Proj}_x(.)$ projects the combined space $\mathbb{R}^{n+m}$ onto the original space $\mathbb{R}^n$, $S(\mathbf{p})$ is the affine subspace:

$$S(\hat{\mathbf{p}}) = \left\{ \begin{pmatrix} \mathbf{x} \\ \mathbf{p} \end{pmatrix} \in \mathbb{R}^{n+m} \middle| \, \mathbf{p} = \hat{\mathbf{p}} \right\} \tag{18}$$

and $F_i^m(\tilde{P}')$ is a m-face of $\tilde{P}'$ found as the intersection between $\tilde{P}'$ and the supporting hyperplanes [8].

For each face of the polyhedron $\tilde{P}'$, a set of active constraints is well defined, resulting in the fact that each point $\begin{pmatrix} \mathbf{x}_i(\mathbf{p})^{\mathrm{T}} & \mathbf{p}^{\mathrm{T}} \end{pmatrix}^{\mathrm{T}} \in F_i^m(\tilde{P}')$ lies in a subspace of dimension m and thus $\mathbf{x}$ and $\mathbf{p}$ are related by:

$$\begin{bmatrix} \mathbf{A}_{eq} \\ \bar{\mathbf{A}}_{in_i} \end{bmatrix} \mathbf{x} = \begin{bmatrix} \mathbf{A}'_{eq} \\ \bar{\mathbf{A}}'_{in_i} \end{bmatrix} \mathbf{p} + \begin{bmatrix} \mathbf{b}_{eq} \\ \bar{\mathbf{b}}_{in_i} \end{bmatrix} \tag{19}$$

where $\bar{\mathbf{A}}_{in_i}, \bar{\mathbf{A}}'_{in_i}, \bar{\mathbf{b}}_{in_i}$ are the subset of the inequalities defined previously, satisfied by saturation. If the matrix $\begin{bmatrix} \mathbf{A}_{eq}^{\mathrm{T}} & \bar{\mathbf{A}}_{in_i}^{\mathrm{T}} \end{bmatrix}^T$ is not invertible, it corresponds to faces $F_i^m(\tilde{P}')$ where for one given $\mathbf{p}$ more than one point $\mathbf{x} \in \Re^n$ is feasible and such combinations do not match a vertex of $P'(\mathbf{p})$ and are the zones where $P'(\mathbf{p})$ changes its shape. In the invertible case, the dependencies are:

$$\mathbf{x}_i(\mathbf{p}) = \begin{bmatrix} \mathbf{A}_{eq} \\ \bar{\mathbf{A}}_{in_i} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{A}'_{eq} \\ \bar{\mathbf{A}}'_{in_i} \end{bmatrix} \mathbf{p} + \begin{bmatrix} \mathbf{A}_{eq} \\ \bar{\mathbf{A}}_{in_i} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{b}_{eq} \\ \bar{\mathbf{b}}_{in_i} \end{bmatrix} \tag{20}$$

For the implementation of these theoretical results, an enumeration of the m-faces must be available. If the projection of this face on the parameters space is a polyhedron of dimension less than m, then the already mentioned condition of an m-face that does not define a vertex of $P'(p)$ is active. Otherwise the projection corresponds to the validity domain VD of the parameterized vertex given by (20).

B. Explicit solution of multiparametric LP (MPLP)

Recalling the problem to be solved similarly to (11):

$$\begin{aligned} x^*(\mathbf{p}) &= \min_{\mathbf{x}} f^T\mathbf{x} \\ \text{subject to } & A_{in}\mathbf{x} \leqslant B_{in}\mathbf{p} + b_{in} \end{aligned} \tag{21}$$

one can use the geometrical description of the feasible domain in terms of a parameterized polyhedron in order to construct the explicit solution. The result (which reveals also the piecewise affine dependence on the parameters) is resumed by the following proposition, (a generalization for parameterized polyhedra of the geometrical description of an LP given in relation with the Chernikova algorithm [7]):

Proposition 2: The solution of a MPLP optimization problem is characterized by the followings:

a) If there exists a bidirectional ray $\mathbf{z}$ such that $f^T\mathbf{z} \neq 0$ or a unidirectional ray $\mathbf{y}$ such that $f^T\mathbf{y} \leqslant 0$, then the minimum is unbounded;

b) For the subdomains of the parameter space $D_{ifez} \in \mathbb{R}^n$ where the associated polyhedron $P = \{\mathbf{x}|A_{in}\mathbf{x} \leqslant B_{in}\mathbf{p} + b_{in}, \mathbf{p} \in D_{ifez}\}$ is empty, the problem is infeasible;

c) All bidirectional rays $\mathbf{z}$ are such that $f^T\mathbf{z} = 0$ and all unidirectional rays y are such that $f^T\mathbf{y} \geqslant 0$. For the sub domains D_k where the minimum:

$$\min\left\{f^T\mathbf{x}_i(\mathbf{p})|\mathbf{x}_i(\mathbf{p}) \text{ vertex of P}(\mathbf{p})\right\}$$

is attained by a constant subset of vertices of $P(\mathbf{p})$, the complete solution is:

$$\begin{aligned} S_k(\mathbf{p}) &= conv.hull\left\{\mathbf{x}^*_{1k}(\mathbf{p}), \ldots, \mathbf{x}^*_{sk}(\mathbf{p})\right\} + \\ &+ cone\left\{\mathbf{y}^*_1, \ldots, \mathbf{y}^*_r\right\} + lin.spaceP(\mathbf{p}) \end{aligned}$$

where $\mathbf{x}^*_i$ are the vertices corresponding to the minimum and $\mathbf{y}^*_i$ are such that $f^T\mathbf{y}^*_i = 0\square$

Adapting this result to our goal of finding the explicit solution for the LP in robust GPC yields to:

Proposition 3: The solution of a MPLP within robust GPC satisfies:

a) The problem is infeasible for the subdomains $D_{ifez} \in \mathbb{R}^n$ where no parameterized vertex is available;

b) Subdomains D_k are defined for the zones where the solution $S_k(\mathbf{p}) = conv.hull\left\{\mathbf{x}^*_{1k}, \ldots, \mathbf{x}^*_{sk}\right\}$ is given by the same set of parameterized vertices satisfying:

$$\begin{aligned} & f^T\mathbf{x}^*_{1k} = \ldots = f^T\mathbf{x}^*_{sk} = \\ & = \min\left\{f^T\mathbf{x}_i(\mathbf{p})|\mathbf{x}_i(\mathbf{p}) \text{ vertex of P}(\mathbf{p})\right\} \end{aligned}$$

Remark: As the parameters in (21) vary in the parameter space, the vertices of the optimization domain may split, shift or merge. The optimum will follow this evolution in the parameter space as it is a continuous function of parameters.

From a practical point of view the implementation of this result is direct and follows the steps:

1) *Find the expression of the parameterized feasible domain in the augmented data+parameter space:*

$$A_{in}\mathbf{x} \leqslant B_{in}\mathbf{p} + b_{in} \Leftrightarrow \begin{bmatrix} \mathrm{A_{in}} & -B_{in} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{p} \end{bmatrix} \leqslant b_{in}$$

2) *Find the m-faces where m is the dimension of the parameter space.*

3) *Retain only those corresponding to parameterized vertices by ignoring those with non-invertible projection on the parameter space*

4) *Compute validity domain D_k for each parameterized vertex*

5) *Compare each pair of vertices. In the case of a non-empty intersection of their validity domains split them using the linear cost function. The final expression will be a union of regions corresponding to the parameterized vertices characterizing the optimum.*

Remark: The difference of two convex domains is not a close operation and the output of the procedure is a union of convex sub-domains of the parameters' space, not necessary covering entire $\mathbb{R}^m$ (*step 4*).

From the robust GPC point of view, the difference:

$$\aleph = \Re^m \backslash \left\{\cup D_k;\ k = 1..n_D\right\} \tag{22}$$

describes the regions of infeasible parameters.

A special attention must be given to the step *5* with the iterative comparison of the vertices and their validity domains. A possible routine may be based on the following procedure:

procedure *CutDomains* (VD: the set of validity domains)
n=cardinal (VD); i=1; j=2;
while $i < n+1$
  while $j < n+1$
    if $VD_j \cap VD_i \neq \emptyset$
      if $f^Tx_i \leqslant f^Tx_j$ *then* $VD_j = VD_j - VD_i$
        else $VD_i = VD_i - VD_j$
      endif
      $j = j + 1;$
    endif
  end
  $i = i + 1$
end

C. Explicit robust GPC law in RST form

The result of the presented algorithm will be a set of r validity domains and for each such domain a linear function can be designed as corresponding to the analytical solution of the optimization (11). The overall function will be piecewise linear and continuous and it can be represented by a look-up table (Table I).

Remark: In the case of the RGPC law, the parameters are the past inputs, past outputs and future setpoints and thus the control law for each validity domain can be rewritten as a polynomial law in the delay operator q^{-1}.

Indeed, one can recall from section II, the fact that the parameters vector was defined as:

$$\mathbf{p} = \left[u_{t-1} \cdots u_{t-n_b}\, y_t \cdots y_{t-n_a}\, w_{t+N_1} \cdots w_{t+N_2}\right]^T$$

TABLE I

LOOK-UP TABLE OF EXPLICIT MULTIPARAMETRIC LP SOLUTION

Validity domain	Linear law
$VD_1 : (M_1\mathbf{p} < m_1)$	$x = F_1\mathbf{p} + f_1$
$\dots$	$\dots$
$VD_r : (M_r\mathbf{p} < m_r)$	$x = F_r\mathbf{p} + f_r$

and thus the optimal linear laws in table I can be rewritten as polynomial control laws due to the fact that the GPC law apply effectively only the first control action:

$$u(t) = T(q)\,w(t) - S(q^{-1})\,u(t) - R(q^{-1})\,y(t) + f$$

The complete explicit formulation of the constrained GPC law is stated as a collection of such affine RST laws:

TABLE II

LOOK-UP TABLE OF EXPLICIT RST LAWS

Validity domain	GPC law
$VD_1 : (M_1\mathbf{p} < m_1)$	$u_t = T_1 w_t - S_1 u_t - R_1 y_t + f_1$
$\dots$	$\dots$
$VD_r : (M_r\mathbf{p} < m_r)$	$u_t = T_r w_t - S_r u_t - R_r y_t + f_n$

Once this look-up table is available, an efficient positioning mechanism can be constructed such that the following on-line routine can find the control action according to the robust GPC philosophy.

Algorithm (on-line solver)

1. Find the corresponding set $D_k, k = 1..r$ for the current $\boldsymbol{p}$. Return infeasible if no D_k found.

2. Compute u_{GPC} using the RST law for D_k.

3. Update the set of context parameters $\boldsymbol{p}$.

4. Restart from 1 with the new $\boldsymbol{p}$.

IV. EXAMPLE

Consider a benchmark system for the positioning control of asynchronous machines [10]. The predictive control law will be designed starting from a model of the mechanical and electrical part together with the zero order hold and the sampler - Fig. 1. However, the influence of the electrical part can be neglected, compared with the mechanical time constant. The mechanical part is characterized by the motor inertia J, the friction coefficient f and Γ the load torque. The discrete time transfer function between the electromechanical torque and the angular displacement for a sampling time $T_e = 14 \times 76.6\mu s = 1,0724ms$ is given by:

$$\frac{\theta(q^{-1})}{u(q^{-1})} = \frac{q^{-1}B(q^{-1})}{A(q^{-1})} = \frac{10^{-4}(0.821q^{-1} + 0.8206q^{-2})}{(1 - q^{-1})(1 - 0.998q^{-1})}.$$

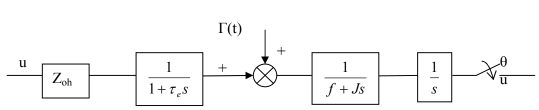


Fig. 1. Model for the MPC design.

In order to validate different controllers and their capabilities, a position benchmark cycle is used, where the speed reference increases to the nominal speed and descends to zero with different profiles. This cycle enables to test the position loop behavior at very slow and zero speeds at nominal speed and during parabolic position tracking.

Using classical GPC procedures (quadratic cost functions), in the unconstrained case, a two degree of freedom RST controller can be designed as in Fig. 2.

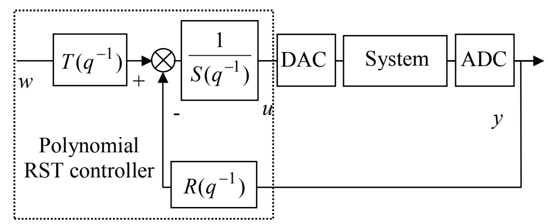


Fig. 2. RST controller scheme.

The time-domain simulation of such a control law based on a prediction horizon of $N_2 = 5$, $N_u = 2$ and a end-point constraint can be visualized in Fig. 3.

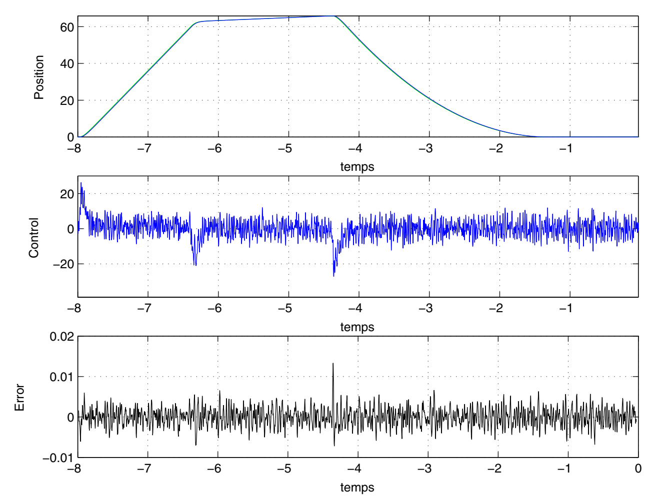


Fig. 3. Classical GPC controller.

The same control idea can be used for the constrained (on the overshoot) case:

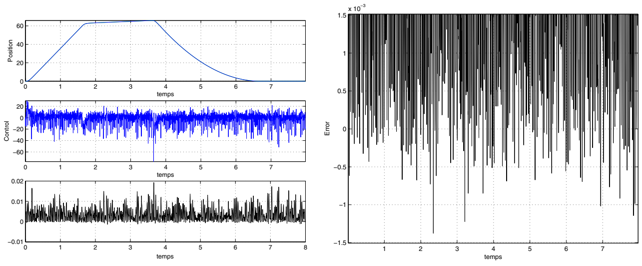
$$y_{t+k} \leqslant w_{t+k}, \forall 1 \leqslant k \leqslant N_2$$

but it doesn't offer a robust constraint satisfaction for disturbances affecting the model:

$$A(q^{-1})\,y_t = B(q^{-1})\,u_{t-1} + v_t$$

$$-0.001 \leqslant v_t \leqslant 0.001$$

The closed loop system with the classical GPC is far from being optimal (Fig. 4a). The tracking error is important and more than that, it is not always positive (Fig. 4b) as demanded by the constraints stated previously. The solution is to implement a robust GPC scheme as presented in the theoretical part of this article. By applying the parameterized polyhedra approach presented earlier, one can obtained for $N_2 = 4; N_u = 1$ an explicit GPC law expressed by a



(a) Position,control and error (b) Zoom on the error

Fig. 4. Classical GPC with constraints

piecewise linear law of 4 RST polynomials in the backward shift operator.

The validity domains of these control laws are unions of convex sets in the space of parameters. An interesting aspect, in this case, is that the RST laws are not containing affine terms. This is due to the fact that the inequality constraints are formulated exclusively on previous inputs, previous outputs and future references and no constant terms interfere. The time simulation confirms the improvement of the closed loop behavior as it can be seen in Fig. 5 (there is no overshoot, the constraints are robustly fulfilled).

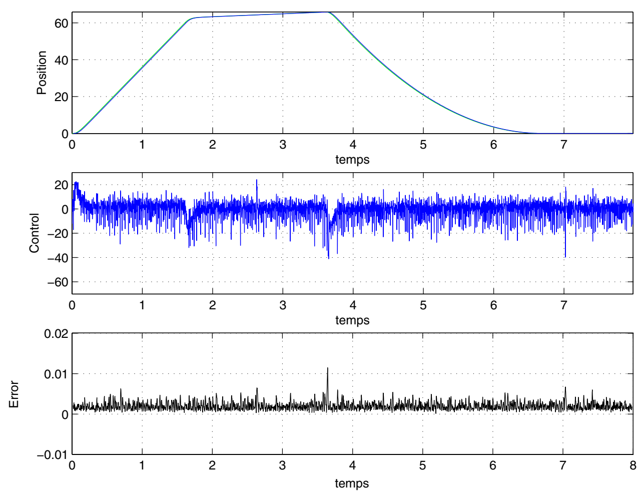


Fig. 5. Robust GPC law with constraints. Prediction horizon $N_2 = 4$.

If the prediction horizon is augmented to $N_2 = 14$ one can observe that the control signal is less sensitive to the presence of disturbances. This is obtained with a deterioration of the tracking error (Fig. 6) which represents a classic compromise in the robust control theory.

V. Conclusions

The paper presented a geometrical approach for the robust Generalized Predictive Control under constraints, confirming the formulation of the optimal sequence as a solution of a multi-parametric linear problem.

The explicit solution of this problem was synthesized by means of parameterized polyhedra. This approach proposes an alternative to the recent methods presented in the literature

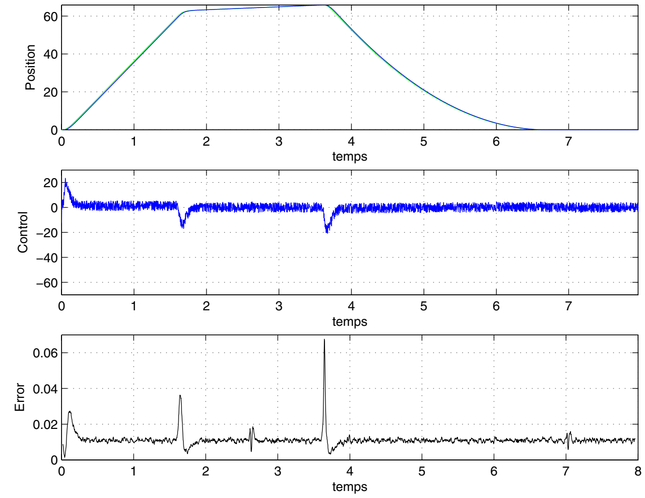


Fig. 6. Robust GPC law with constraints. Prediction horizon $N_2 = 14$.

with an insight on the geometry of the feasible set. Its advantages might be the fact that optimum lies on the parameterized vertices providing a direct constant linear affine dependence in the context parameters. The disadvantages are linked to the fact that the procedure of identifying the parameterized vertices might be computationally demanding for long prediction horizons even if on-line procedures.

References

[1] A. Bemporad, M. Morari, V. Dua, E. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems", *Automatica* vol.38, 2002, pp. 3-20.

[2] A. Bemporad, F. Borelli and M. Morari. "Robust Model Predictive Control: Piecewise Linear Explicit Solution". *Proc. European Control Conference*, pp. 939-944, Karlsruhe, 2001.

[3] E.F. Camacho and C. Bordons, *Model Predictive Control*, Springer, 2003.

[4] G.C. Goodwin, M.M. Seron and J.A. De Dona. *Constrained Control and Estimation*. Springer-Verlag, London, 2004.

[5] E.C. Kerrigan, and J.M. Maciejowski. "Feedback min-max model predictive control using a single linear program: robust stability and the explicit solution ". *International Journal of Robust and Nonlinear Control* , vol. 14-4, 395-413, 2004.

[6] M. V. Kothare, P.J. Campo, M. Morari, and C. N. Nett. "A unified framework for the study of anti-windup designs", *Automatica*, vol. 30, pp. 1869-1883, 1994.

[7] H. Leverge. "A Note on Chernikova's Algorithm". *Technical Report 635*, IRISA, France, 1994.

[8] V. Loechner, and D.K. Wilde. "Parameterized Polyhedra and their Vertices". *Int. Journal of Parallel Programming*, 25(6), December, 1997.

[9] D.Q. Mayne, J.B. Rawlings, C.V. Rao, and P.O.M. Scokaert, "Constrained model predictive control: Stability and optimality", *Automatica*, Vol.36, pp. 789-814, 2000.

[10] E. Mendes, and J.P. Barbot, "Benchmark transitique rapide", *APII Journal Europen des Systmes Automatiss*, Vol. 36-5, pp. 701-708, 2002.

[11] S. Olaru and D. Dumur, "A Parameterized Polyhedra Approach for Explicit Constrained Predictive Control", *43rd IEEE Conference on Decision and Control*, Bahamas, 2004.

[12] P.O.M. Scokaert, and D.Q. Mayne. "Min-max Feedback Model Predictive Control for Constrained Linear Systems". *IEEE Trans. Automatic Control*, 43-8, pp. 1136-1142, 1998.