Proceedings Paper:

Dughman, S.S. and Rossiter, J.A. orcid.org/0000-0002-1336-0633 (2017) The feasibility of parametric approaches to predictive control when using far future feed forward information. In: 2017 13th IEEE International Conference on Control & Automation (ICCA). 13th IEEE International Conference on Control and Automation, 03/07/2017 - 06/07/2017, Ohrid, Macedonia. Institute of Electrical and Electronics Engineers .

https://doi.org/10.1109/ICCA.2017.8003215

The feasibility of parametric approaches to predictive control when using far future feed forward information

S. S. Dughman
Department of Automatic Control
and Systems Engineering
University of Sheffield
Sheffield S1 3JD UK
Email: ssdughman1@sheffield.ac.uk

J. A. Rossiter
Department of Automatic Control
and Systems Engineering
University of Sheffield
Sheffield S1 3JD UK
Email: j.a.rossiter@sheffield.ac.uk

***Abstract*—This paper considers the tractability of parametric solvers for predictive control based optimisations, when future target information is incorporated. it is shown that the inclusion of future target information can significantly increase the implied parametric dimension to an extent that is undesirable and likely to lead to intractable problems. The paper then proposes some alternative methods for incorporating the desired target information, while minimising he implied growth in the parametric dimensions, at some possibly small cost to optimality.**

***Index Terms*—Parametric predictive control, advance knowledge, computational efficiency.**

I. Introduction

One of the most significant advances in predictive control of the past 20 years has the been the recognition that one can define the solution of a quadratic programme (QP), in full, using off-line computations [1], [12]. As long as this off-line (or so called parametric) solution is not to complex, then coding and implementing this on-line may be far simpler than implementing an on-line QP optimiser. The parametric solution offers the potential for reliability, transparency (important for validation and certification) and most importantly, very fast sample rates for some systems.

Nevertheless, parametric solutions also have their disadvantages and the literature is full of possible solutions to counter these [10]. For example: (i) in some cases the parametric solution can be difficult to compute reliably due to poor conditioning; (ii) where the parametric solution requires large numbers of regions it may not longer be computationally efficient.

This brief paper makes a minor contribution to one aspect of computational complexity. To define what this contribution will be, it is first useful to define a generic QP optimisation and its parametric solution.

$$\min_{z} z^T S z + z^T P w \quad \text{s.t.} \quad Nz + Mw \leq d \tag{1}$$

where w is a system state and z are the degrees of freedom (d.o.f.) and parameters S, P, N, M, d define the cost function and linear constraints. A parametric solution (often denoted MPQP) partitions the space into a number of non-overlapping regions for the system state such that the optimal solution for (1) is equivalent to:

$$N_i w \leq d_i \quad \Rightarrow \quad z = K_i w + p_i, \quad i = 1, 2, \cdots, n \tag{2}$$

for suitable N_i, d_i, K_i, p_i.

It is recognised that there is a strong link between the dimension of the state w and the required number of regions n to capture the entire solution. Hence, in general, parametric solutions are favoured for systems with a low state dimension but less likely to be useful for high state dimensions. Moreover, the higher the state dimension, the more likely one is to encounter conditioning problems in the MPQP solver. The main aim of this paper therefore is ask the question, can we reduce the state dimension for some specific predictive control problems? In particular, the focus here is on the handling of feedforward information such as future target information which, in principle, can be embedded systematically into predictive control algorithms.

Section 2 will outline the predictive control problems to be discussed and sections 3 will demonstrate how the state dimension can be reduced using some elementary algebra and re-parametrisation for the optimisation degrees of freedom. The paper finishes with some examples and conclusions.

II. Background on predictive control

This section summarises briefly the optimisation implicit in two popular predictive control algorithms, generalised predictive control (GPC) [4] and optimal predictive control (OMPC) [14], [18]; some fine details are omitted as both algorithms are well known and some details are not central to this paper. A specific and important point is to consider how set point information is incorporated [5], [6], [19] as this detail is often omitted or neglected in the mainstream literature.

For simplicity this paper assumes a state space model of the following form.

$$x_{k+1} = Ax_k + B\Delta u_k; \quad y_k = Cx_k \tag{3}$$

with x_k, u_k, y_k the states, input and output respectively with dimensions n_x, m, m and $\Delta u_k = u_k - u_{k-1}$. The system is subject to constraints, typically (others are possible):

$$\underline{u} \le u_k \le \overline{u}; \quad \underline{\Delta u} \le \Delta u_k \le \overline{\Delta u}; \quad \underline{y} \le y_k \le \overline{y} \tag{4}$$

Furthermore, define the future target $\underset{\to k+1}{r}$ as follows:

$$\underset{\to k+1}{r} = [r_{k+1}^T, r_{k+2}^T, \cdots, r_{k+n_y}^T]^T \tag{5}$$

A. Generalised predictive control [4], [15]

GPC uses a finite horizon performance index of the following form:

$$J = \|\underset{\to k+1}{r} - \underset{\to k+1}{y}\|_2^2 + \lambda \|\Delta \underset{\to}{u}\|_2^2 \tag{6}$$

where the definitions of $\underset{\to k+1}{y}, \Delta \underset{\to k}{u}$ are analogous to (5) except that $\Delta \underset{\to k}{u}$ has n_u terms with $n_u \ll n_y$ in general. The predictions can be shown to obey equations of the following form, for suitable matrices P_x, H:

$$\underset{\to k+1}{y} = H \Delta \underset{\to k}{u} + P_x x_k \tag{7}$$

Expanding the performance index in full using the predictions of (7) gives:

$$\begin{array}{l} J = \Delta \underset{\to k}{u}^T (H^T H + \lambda I) \Delta \underset{\to k}{u} + \Delta \underset{\to k}{u}^T H^T (P_x x_k - \underset{\to k+1}{r}) \\ \quad +\alpha; \qquad \alpha = (\underset{\to k+1}{r} - P_x x_k)^T (\underset{\to k+1}{r} - P_x x_k) \end{array} \tag{8}$$

Satisfaction of constraints (4) over the horizon n_y can be shown to be equivalent to a set of linear inequalities captured as follows (for suitable M, N, d):

$$N \Delta \underset{\to k}{u} + M x_k \le d \tag{9}$$

Note, it is implicit here that the model (3) includes a state with information about u_k.

Algorithm 2.1: GPC is defined as follows. At each sample, perform the quadratic programming optimisation

$$\min_{\Delta \underset{\to k}{u}} J \quad \text{s.t.} \quad N \Delta \underset{\to k}{u} + M x_k \le d \tag{10}$$

Implement the first value of $\Delta \underset{\to k}{u}$, that is Δu_k.

Next a lemma and corollary illustrate the major difficulty which is the focus of this paper.

Lemma 1: Optimisation (10) can be recast in the same form as (1).

Proof: Removing the term α which does not depend upon the degrees of freedom, and then combining parameters which may vary with time (that is $\underset{\to k+1}{r}, x_k$) the performance index can be reformatted as:

$$J \equiv \Delta \underset{\to k}{u}^T \underbrace{(H^T H + \lambda I)}_{S} \Delta \underset{\to k}{u} + \Delta \underset{\to k}{u}^T \underbrace{H^T [P_x, -I]}_{P} \underbrace{\begin{bmatrix} x_k \\ \underset{\to k+1}{r} \end{bmatrix}}_{w_k} \tag{11}$$

In a similar way, the constraints (9) can be reformatted as follows:

$$N \Delta \underset{\to k}{u} + [M, 0] w_k \le d \tag{12}$$

□

Corollary 1: Including advance knowledge $\underset{\to k+1}{r}$ in the performance index augments the implied parametric state dimension by the dimension of $\underset{\to k+1}{r}$ to give a dimension of $n_x + m \times n_y$. This is obvious from the replacement of x_k by w_k in (11,12).

Now we are in a position to state the obvious dilemma. Where a user would like to incorporate advance knowledge of the target into their predictive control problem, this massively increases the dimension of the parametric state w_k. Even just including a fixed target and integral action [17] increases the required parameter dimension to $n_x + m$ which is already undesirable, to include more advance information could make the dimension of the parametric optimisation impractical in general.

Hence one objective of this paper is to suggest ways of modifying optimisation (11,12) so that one retains some of the benefits of including advance information while at the same time keeping the dimension of the implied parameter space small.

B. Optimal or dual-mode predictive control

GPC is a finite horizon approach which, if tuned carefully, can be effective. However, it is now well known [11] that dual-mode approaches have better properties in general. Consequently, it is worth considering how a parametric solution can be determined for a dual-mode approach. For convenience the standard OMPC algorithm of [14] is utilised.

OMPC uses a infinite horizon performance index of the following form:

$$\begin{array}{l} J = \sum_{i=0}^{\infty} (x_{k+1+i} - x_{ss})^T Q (x_{k+1+i} - x_{ss}) \\ \qquad + (u_{k+i} - u_{ss})^T R (u_{k+i} - u_{ss}) \end{array} \tag{13}$$

along with an input parametrisation of the form:

$$\begin{array}{ll} u_{k+i} - u_{ss} = -K(x_{k+i} - x_{ss}) + c_{k+i} & i = 0, 1, \cdots, n_c - 1 \\ u_{k+i} - u_{ss} = -K(x_{k+i} - x_{ss}) & i \ge n_c \end{array} \tag{14}$$

so the variables c_{k+i} are the degrees of freedom and (x_{ss}, u_{ss}) are the expected steady states to track a fixed target r_{k+1}; typically [15] one can show that $x_{ss} = T_1 r_{k+1},\ u_{ss} = T_2 r_{k+1}$ for suitable T_1, T_2. Substituting from (14) into (13), minimising the performance index can be shown to be equivalent [14] to minimising the following form:

$$J = \underset{\to k}{c}^T S \underset{\to k}{c} \tag{15}$$

Combining model (3), input predictions (14) and constraints (4) inequalities representing constraint satisfaction of the predictions can be reduced to:

$$N \underset{\to k}{c} + M x_k + Q r_{k+1} \le d \tag{16}$$

for suitable N, M, Q.

Algorithm 2.2: OMPC is defined as follows. At each sample, perform the quadratic programming optimisation

$$\min_{\underset{\to k}{c}} \underset{\to k}{c}^T S \underset{\to k}{c} \quad \text{s.t.} \quad N \underset{\to k}{c} + M x_k + Q r_{k+1} \le d \tag{17}$$

Implement the first value of c_k in (14) to determine the current input, that is u_k.

Lemma 2: Optimisation (17) can be recast in the same form as (1). This is obvious.

Corollary 2: Extension of dual-mode strategies to take account of more future values of the target such as available in $\underset{\rightarrow k+1}{r}$ is not widely discussed in the literature [5]. In aid of brevity, here it will simply be noted that the performance index and inequalities and thus required optimisation can take the following form:

$$\begin{array}{rl} \min_{\underset{\rightarrow k}{c}} & \underset{\rightarrow k}{c}^T S \underset{\rightarrow k}{c} + \underset{\rightarrow k}{c}^T S_2 \underset{\rightarrow k+1}{r} \\ \text{s.t.} & N \underset{\rightarrow k}{c} + [M, Q] \begin{bmatrix} x_k \\ \underset{\rightarrow k+1}{r} \end{bmatrix} \leq d \end{array} \tag{18}$$

As in the previous section, it is noted that the incorporation of advance information has vastly increased the dimension of the associated parametric optimisation. In essence, for the equivalent of optimisation (1), and exactly as in the previous subsection, the implied state is now $w = [x^T, \underset{\rightarrow k+1}{r}^T]^T$.

C. Summary and proposals

It has been shown that a simplistic inclusion of future target information $\underset{\rightarrow k+1}{r}$ into a predictive control algorithm leads to an increase in the dimension of the parametric space for the associated multi-parametric quadratic programme. In general, for anything other than the most trivial case [17] where it is assumed that $r_{k+i} = r_{k+1}, \forall i$ so that the effective dimension of $\underset{\rightarrow k+1}{r}$ reduces to that of just r_{k+1} , then this increase in dimension is likely to be unmanageable and thus a parametric approach is unlikely to be feasible. In consequence, this paper considers for which scenarios can this information be incorporated without leading to unnecessarily large dimensional increases.

III. Reducing the dimension of the parameter space with finite horizon algorithms

This section will show how small changes to the formulation of a GPC optimisation can reduce the dimension of the implied parameter space. Some suggestions lead to a small degree of sub-optimality, but in fact, within parametric predictive control, the use of sub-optimality is often a key tool for reducing complexity [2], [3] and thus this may be considered an acceptable compromise in order to gain some of the benefits of using the feedforward information rather than ignoring it.

A. Reducing the amount of advance knowledge

Recent work on the use of advance information [6] within predictive control has considered questions about how much advance information is useful, that is, really makes a noticeable difference to closed-loop performance. It was established that ignoring far future (beyond n_a samples) values of the target usually led to a minimal deterioration in performance as long as $n_u < n_a < (n_u + n_r)/2$ with n_r being the notional rise time. Larger n_a were usually unhelpful as the control d.o.f. were not contemporaneous enough and therefore inappropriate control moves for the relevant target changes. Therefore, as in the context of parametric approaches some sub-optimality is accepted in the pursuit of simplicity, this section looks at what can be achieved by summarising future target information $\underset{\rightarrow k+1}{r}$ into fewer values

The most obvious and easiest way to reduce the dimension of the parameter space in vector w is the rather obvious one of reducing the dimension of $\underset{\rightarrow k+1}{r}$. It is commonplace in the predictive control field to use the following approximation, assume $r_{k+n_a+i} = r_{k+n_a}, \forall i > 0$ (here given in SISO case to simplify algebra). Then:

$$\underset{\rightarrow k+1}{r} = \begin{bmatrix} r_{k+1} \\ \vdots \\ r_{k+n_a} \\ \vdots \\ r_{k+n_a} \end{bmatrix} = \begin{bmatrix} r_{k+1} \\ r_{k+2} \\ \vdots \\ r_{k+n_a-1} \\ L_{n_y-n_a} r_{k+n_a} \end{bmatrix}; \; L_{n_y-n_a} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \tag{19}$$

where L_p is a $p-$dimensional vector of ones. This assumption reduces the dimension of $\underset{\rightarrow k+1}{r}$ from n_y components to n_a components. Moreover, it has been shown [5], [19] that in many cases using relatively small values of n_a give almost equivalent, and sometimes better, closed-loop performance compared to using values close to n_y. Thus using the approximation implicit in a choice of small n_a is reasonable for GPC.

Remark 1: Even though one can reduce the overall parameter dimension of $w = [x^T, \underset{\rightarrow k+1}{r}^T]^T$ to $n_x + n_a \times m$ with $n_a < n_y$, one might still argue that anything much beyond $n_a = 2$ is likely to increase the parameter space beyond normally accepted limits for parametric solutions. While $n_a = 2$ usually gives better closed-loop performance than $n_a = 1$, nevertheless it may still be significantly worse performance than achievable with an even larger n_a and thus such a solution may not be sufficient in general.

B. Reducing the amount of advance knowledge further

Existing literature has largely focussed on the structure of (19) and argued that $n_a \ll n_y$ often leads to improved closed-loop behaviour [5], [19]. However, there is another alternative that has not been explored carefully and is the subject of a current investigation. Another interesting avenue is the extent to which transient values such as r_{k+1}, r_{k+2} are really useful as most systems cannot respond significantly within a few samples, hence having a particular target during fast transients may not be meaningful. The proposal here therefore is to ignore specific information about the targets for the next few samples and instead assume that $r_{k+i} = r_{k+n_{a2}}, i \leq n_{a2}$ and thus use a structure such as the follows.

$$\underset{\rightarrow k+1}{r} = \begin{bmatrix} r_{k+n_{a2}} \\ \vdots \\ r_{k+n_{a2}} \\ \vdots \\ r_{k+n_a} \\ \vdots \\ r_{k+n_a} \end{bmatrix} = \begin{bmatrix} L_{n_{a2}} r_{k+n_{a2}} \\ r_{k+n_{a2}+1} \\ \vdots \\ r_{k+n_a-1} \\ L_{n_y-n_a} r_{k+n_a} \end{bmatrix} \tag{20}$$

It is clear that the dimension of the corresponding vector $\underset{\rightarrow k+1}{r}$ has now been reduced to having $n_a - n_{a2} + 1$ independent components, which is a significant reduction compared to n_y components.

C. Using insights from reference governors and PFC

Reference governors [8] are primarily focussed on highly efficient constraint handling whereby one ensures that the target to the feedback loop changes slowly enough not to cause the internal signals to violate constraints. To some extent, performance takes second place to computational efficiency and simplicity so some sub-optimality is accepted. In the context of this paper, a key observation is the use of small amounts of feed forward information of the target rather than the entire trajectory. Specifically, this paper notes one possible simplification which is implicit in PFC [13], that is assume the future target trajectory takes the following form (a smooth transition from current output to long term target):

$$r_{k+i} = (r_{k+n_a} - y_k)(1-\lambda^i) + y_k; \quad \underset{\rightarrow k+1}{r} = W_1 r_{k+n_a} + W_2 y_k \tag{21}$$

where the definitions of W_1, W_2 are obvious and r_{k+n_a} is the best representation of the long term target value.

Clearly, this suggestion has close analogies to the previous two subsections in that the future target information is approximated in some fashion to reduce the dimension of the implied parametric space. The proposal here has the advantage that the parametric space is the same dimension as would be needed for routine inclusion of integral action [17], although of course the use of the feedforward information is now much less precise than it could be due to the approximation implicit in (21).

Remark 2: All three suggestions in the previous subsections reduce to the following generic approximation.

$$\underset{\rightarrow k+1}{r} = W_1 \gamma + W_2 x_k \tag{22}$$

where γ constitutes the degrees of freedom to encapsulate future values of r_{k+i} and W_1, W_2 are defined appropriately. In consequence, the parametric dimension required to include future target information is exactly the dimension of γ. As noted earlier, to include integral action [17], at the very least this must match the output dimension.

D. Utilising the unconstrained optimal

A key observation with finite horizon algorithms such as GPC is that the target values $\underset{\rightarrow k+1}{r}$ do not appear in the constraint set (12). Consequently, if one can reduce the implied dimension of w in the cost function J, then this reduction in parameter space will apply for the QP as a whole.

One obvious mechanism for altering a QP optimisation is to re-parametrise relative to the unconstrained optimal [14], [16], [18]. Consequently, define the unconstrained optimal as follows:

$$\Delta \underset{\rightarrow nom}{u} = -(H^T H + \lambda I)^{-1} H^T [P_x, -I] w_k \tag{23}$$

Next, write the actual future inputs as deviations from the unconstrained optimal.

$$\Delta \underset{\rightarrow k}{u} = \Delta \underset{\rightarrow nom}{u} + \Delta \underset{\rightarrow}{\tilde{u}} \tag{24}$$

Finally, substitute (24) into (11) and hence:

$$J = (\Delta \underset{\rightarrow nom}{u} + \Delta \underset{\rightarrow}{\tilde{u}})^T S (\Delta \underset{\rightarrow nom}{u} + \Delta \underset{\rightarrow}{\tilde{u}}) + (\Delta \underset{\rightarrow nom}{u} + \Delta \underset{\rightarrow}{\tilde{u}})^T P w_k \tag{25}$$

Lemma 3: The minimisation of J in (25) is equivalent to the minimisation of the following performance index.

$$J = (\Delta \underset{\rightarrow}{\tilde{u}})^T S (\Delta \underset{\rightarrow}{\tilde{u}}) \tag{26}$$

Proof: This follows directly from $\Delta \underset{\rightarrow nom}{u}$ being the unconstrained optimal. Therefore in the unconstrained case, the optimal value of $\Delta \underset{\rightarrow}{\tilde{u}} = 0$, and therefore a cost written in terms of $\Delta \underset{\rightarrow}{\tilde{u}}$ cannot have a linear term as this would imply a non-zero unconstrained optimum. □

Lemma 4: The parametrisation of (24) modifies the inequalities of (12) as follows. Proof omitted as obvious.

$$N \Delta \underset{\rightarrow}{\tilde{u}} + [N, M] \begin{bmatrix} \Delta \underset{\rightarrow nom}{u} \\ x_k \end{bmatrix} \leq d \tag{27}$$

Theorem 1: Using parametrisation (24) changes the dimension of the parameter space from $n_x + n_y \times m$ to $n_x + n_u \times m$.

Proof: This is obvious from (25,27) as the parameter $\Delta \underset{\rightarrow nom}{u}$ has dimension $n_u \times m$ and the constraints also include parameter x_k.

Remark 3: Because the future target information can be subsumed into the unconstrained optimal, if $n_u < n_a$ then one can reduce the implied parameter space for a MPQP solution below that suggested in subsection III-A. Nevertheless it is still transparently clear that the inclusion of advance knowledge will inevitably lead to a larger increase in the implied parameter space than is likely to be desirable as common tuning guidance suggests that both $n_u \gg 1$ and $n_a \gg 1$. Of course, any form of offset free tracking must as a minimum deploy $n_a = 1$ so all parameter dimensions are relative to that baseline.

E. Re-parametrising the input degrees of freedom

The previous subsections accepted the GPC algorithm in its basic form and asked questions about the implications of adding advance knowledge of targets into an MPQP solver. A key observation was that the dimension of the parameter space can be linked directly to the number of d.o.f. in the optimisation, while still including all far future feedforward information.

This section pursues the alternative route of changing the algorithm at the outset in the hope that this will lead to a simpler MPQP problem and specifically, asks the question whether the number of optimisation d.o.f. can be reduced, thus benefiting from the insights gained in theorem 1. Specifically, consider the potential benefits of orthonormal parametrisations [9], [16], [20] as these have been shown to effective at enabling a reduction in the number of optimisation d.o.f. while retaining good performance.

The basic suggestion [20] is to write the future control moves as follows:

$$\Delta \underset{\rightarrow k}{u} = \underbrace{\begin{bmatrix} l_{1,0} & l_{2,0} & \cdots \\ l_{1,1} & l_{2,1} & \cdots \\ \vdots & \vdots & \vdots \end{bmatrix}}_{H_L} \underbrace{\begin{bmatrix} \eta_1 \\ \vdots \\ \eta_{n_\eta - 1} \end{bmatrix}}_{\eta} \tag{28}$$

where $l_{i,k}$ are coefficients of the expansion of the ith orthonormal function, for example:

$$l_i(z) = l_{i,0} + l_{i,1}z^{-1} + l_{i,2}z^{-2} + \cdots \tag{29}$$

The conjecture is that in many cases, with appropriate choices of orthonormal functions, one can use $n_\eta < n_u$ and still obtain similar performance and thus reduce the computational complexity of the MPC algorithm.

Lemma 5: Using (28) in the cost function (11) one can derive the unconstrained optimal as:

$$\eta_{nom} = -(H_L^T H_L + \lambda I)^{-1} H_L^T [P_x, -I] w_k \tag{30}$$

This is obvious so not proved.

Theorem 2: One can formulate a performance index and constraint inequalities based on deviation variables for η as follows:

$$J = (\tilde{\eta})^T S_L(\tilde{\eta}); \quad N H_L \tilde{\eta} + [N H_l, M] \begin{bmatrix} \eta_{nom} \\ x_k \end{bmatrix} \leq d \tag{31}$$

Proof: This is exactly analogous to the derivations in lemmata (3,4).

In summary, the dimension of the corresponding parameter space in an MPQP problem of (31) is $n_x + m \times n_\eta$ and thus, in cases where $n_\eta < n_u$, this could be advantageous as compared to the approach in section III-D.

IV. Dual-mode approaches

The QP optimisations for dual mode approaches have one notable difference from those for finite horizon approaches which is evident from viewing the inequalities of (12,16). In one case the values of $\underset{\rightarrow k+1}{r}$ appear explicitly within the constraint inequalities whereas in the other they do not. The consequence of this is that while one can do a reparameterisation in the finite horizon case (see section III-D) to capture the impact of future targets on the cost function within a lower dimensional optimal input trajectory, this is not the case in the dual-mode case because $\underset{\rightarrow k+1}{r}$ still exists in its entirety in the constraints.

In consequence, the main options available to a dual-mode approach correspond to those detailed in sections (III-A to III-C). In aid of brevity, these developments are not restated as they will be exactly equivalent, that is one can use an approximation of the form:

$$\underset{\rightarrow k+1}{r} = W_1\gamma + W_2 x_k \tag{32}$$

where γ constitutes the number of degrees of freedom used to approximate target information, with W_1, W_2 defined appropriately.

V. Numerical examples

This section will give a few numerical examples to demonstrate the impact on parametric complexity of including advance knowledge, using OMPC approach.

Consider example with 2 states.

$$A = \begin{bmatrix} 0.8 & 0.1 \\ -0.2 & 0.9 \end{bmatrix}; \ B = \begin{bmatrix} 0.3 \\ 0.8 \end{bmatrix};$$
$$C = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad D = \begin{bmatrix} 0 \end{bmatrix} \tag{33}$$
$$-0.2 \leq u \leq 0.5; \begin{bmatrix} 1 & 0.2 \\ -0.1 & 0.4 \\ -1 & -0.2 \\ 0.1 & -0.4 \end{bmatrix} x_k \leq \begin{bmatrix} 8 \\ 8 \\ 1.6 \\ 5 \end{bmatrix}$$

The parametric solutions are computed with a range of values of n_a, and the following information is captured.

- Number of inequalities.
- Number of parametric regions in solution.

1) Algorithm in section (III-B). Table I shows the impact of the dimension of the reduced future information (γ) on the parametric solution for different R, using Algorithm in section (III-B).
   It is clear from the result that the number of regions and inequalities increase as the the dimension of the reduced future information (γ) increases.
2) Algorithm in section (III-C). Table II shows the impact of the dimension of the reduced future information (γ) on the parametric solution for different R, using Algorithm in section (III-C).
   Similarly, It is clear from the result that the number of regions and inequalities increase as the the dimension of the reduced future information (γ) increases.

VI. Conclusions and future work

First and foremost it is clear that including advance information about targets (and equivalently measureable disturbances) increases the dimension of the parameter space for an MPQP approach to predictive control. It is recognised that MPQP is often impractical for large parameter spaces and thus one may infer that usually MPQP would be difficult to use in conjunction with advance knowledge scenarios.

Nevertheless, this paper has introduced some reformulations of a typical finite horizon MPC algorithm which can, to a

TABLE I

COMPARISON OF MPQP SOLUTION COMPLEXITY FOR DIFFERENT DIMENSIONS OF γ FORALGORITHM OF SECTION (III-B)

System (33) with $n_a = 3$, $R = 0.1 * I$			
Dimension of $\gamma = n_a - n_{a1} + 1$	1	2	3
Number of regions	70	85	97
Number of inequalities for all regions	39	32	42
System 1 with $n_a = 3$, $\lambda = 0, R = I$			
Dimension of $\gamma = n_a - n_{a1} + 1$	1	2	3
Number of regions	91	123	130
Number of inequalities for all regions	30	32	34

TABLE II

COMPARISON OF MPQP SOLUTION COMPLEXITY FOR DIFFERENT DIMENSIONS OF γ FOR ALGORITHM OF SECTION (III-B)

System (33) with $n_a = 3$, $\lambda = 0.2$, $R = 0.1 * I$			
Dimension of $\gamma = n_a - n_{a1} + 1$	1	2	3
Number of regions	75	106	104
Number of inequalities for all regions	41	41	35
System 1 with $n_a = 3$, $\lambda = 0.2, R = I$			
Dimension of $\gamma = n_a - n_{a1} + 1$	1	2	3
Number of regions	91	124	129
Number of inequalities for all regions	31	34	34

limited extent, overcome problems with dimension growth. In the case of finite horizon approaches, one can exploit the fact that target information appears only in the cost function, but not the constraints. Hence, it is shown that even where a moderately large advance information is used, by reformulating the optimisation in terms of deviation variables about about the unconstrained optimal input trajectory, the parameter space increase can be limited to the control horizon or equivalently the number of optimisation variables. However, a second and perhaps more helpful observation is to exploit the 'added value' in the future target information and capture this value in fewer variables; in essence the increase in the parameter space is linked to the number of variables needed to capture the useful information in the target trajectory and, if needed, one can capture this with very few variables and thus reduce the dimension to that required for incorporating integral action (section III-C). Obviously any simplification of the target information results in some suboptimality, but that is likely to be a price worth paying for improvements in the simplicity of the MPQP solution.

While finite horizon MPC algorithms still dominate industrial practice, nevertheless, it is recognised that dual-mode approaches have significant theoretical and potentially performance advantages and thus it is important to consider the extent to which advance target information can be included in parametric solutions for these. However, it is immediately clear that some of the simplifications possible for finite horizon approaches are no longer possible because the embedding of terminal control laws into dual-mode predictions means that the target information appears in both the cost function and the constraints, and thus one cannot exploit the unconstrained optimal, thus the options of sections III-D,III-E are not available. Hence, in this case, the main option is the approximation of $\underset{\rightarrow k+1}{r}$, and thus the parameter space increase is linked directly to the number of variables used to capture the future target information.

REFERENCES

[1] Bemporad, A., Morari, M., Dua, V. and Pistikopoulos, E.N., (2002), The explicit linear quadratic regulator for constrained systems Automatica, 38 (1), pp. 320

[2] Bemporad, A. and Filippi,C. (2006),An algorithm for approximate multiparametric convex programming, Computational optimization and Applications, 35, pp-87-108.

[3] Canale, M., Fagiano, L. and Milanese, M. (2009), Set membership approximation theory for fast implementation of model predictive control laws, Automatica, 45, pp45-54.

[4] Clarke, D.W. and Mohtadi, C. (1989), Properties of generalized predictive control. *Automatica*, 25(6), 859–875.

[5] Dugham, S. and Rossiter, J. (2016), Systematic and simple guidance for feed forward design in model predictive control. *International Conference on Control Science and Systems Engineering*.

[6] Dugham, S. and Rossiter, J. (2016b), Efficient Feed Forward design within MPC, European Control Conference.

[7] Gilbert, E. and Tan, K. (1991), Linear systems with state and control constraints: The theory and application of maximal admissible sets. *IEEE Transactions Automatic Control*, 36, 1008–1020.

[8] Gilbert, E.G. and Kolmanovsky, I. (1995), Discrete time reference governors and the non-linear control of systems with state and control constraints, *Int. J. Robust and non-linear control*, 5, 487-504.

[9] Khan, B. and Rossiter, J.A. (2013), Alternative Parameterisation within predictive Control: a systematic selection, IJC, 86, 1397–1409. *Automatica*, 34(1), 65–73.

[10] Kouramas, K.I., Faisca, N.P., Panos, C. and Pistokopoulos, E.N. (2011), Explicit/multi-parametric model predictive control (MPC) of linear discrete-time systems by dynamic and multi-parametric programming, Automatica, 47, pp1638–1645.

[11] Mayne, D.Q., Rawlings, J.B., Rao, C.C. and Scokaert, P.O. (2000), Constrained model predictive control: feasibility and optimality. *Automatica*, 36, 789–814.

[12] Pistikopoulos, E.N., Dua, V., Bozinis, N.A., Bemporad, A., Morari, M. (2002), On-line optimization via off-line parametric optimization tools Computers and Chemical Engineering, 26 (2), pp. 175185

[13] Richalet, J. and O'Donovan, D. (2009). *Predictive functional control: principles and industrial applications*. Springer.

[14] Rossiter, J., Kouvaritakis, B., and Rice, M. (1998), A numerically robust state-space approach to stable predictive control stratgies. *Automatica*, 34(1), 65–73.

[15] Rossiter, J.A. (2003). *Model-based predictive control: a practical approach*. CRC press.

[16] Rossiter, J.A., Kouvaritakis, B. and Cannon, M. (2005), An algorithm for reducing complexity in parametric predictive control, IJC, 78, 18, pp1511–1520

[17] Sakizlis, V., Perkins, J. D., and Pistikopoulos, E. N. (2003), The explicit model-based tracking control law via parametric programming, International symposium on advanced control of chemical processes (AD-CHEM).

[18] Scokaert, P.O. and Rawlings, J.B. (1998), Constrained linear quadratic regulation. *Automatic Control, IEEE Transactions on*, 43(8), 1163–1169.

[19] Valencia-Palomo, G., Rossiter, J., and López-Estrada, F. (2014), Improving the feed-forward compensator in predictive control for setpoint tracking. *ISA transactions*, 53(3), 755–766.

[20] Wang, L. (2014), Model predictive control system design and implementation using MATLAB, Springer.