

A systematic selection of an alternative parameterisation for predictive control

Bilal Khan

Automatic Control and Systems Engineering
The University of Sheffield, UK
Email: b.khan@sheffield.ac.uk

John Anthony Rossiter

Automatic Control and Systems Engineering
The University of Sheffield, UK
Email: j.a.rossiter@sheffield.ac.uk

Abstract—Alternative parameterisations have been shown to improve the feasible region for a predictive control law when the number of degrees of freedom is limited. One question yet to be resolved is: which alternative parameterisation is best for a particular problem and what choice of parameter(s) within each parameterisation will lead to an improved feasible region with good performance? This paper tackles this question and demonstrates two systematic approaches to select the best alternative parameterisations. These approaches are based on multiobjective optimisation and a pragmatic selection. Numerical examples demonstrate the efficacy of both methods.
Keywords: Alternative parameterisation, Feasibility, multiobjective optimisation.

I. INTRODUCTION

Model Based Predictive Control (MPC) or receding horizon control (RHC) or embedded optimisation or moving horizon or predictive control [2], [3], [1], is the general name for different computer control algorithms that use past information of the inputs and outputs and a mathematical model of the plant to optimise predicted future behaviour. MPC has been developed widely both in the process industry and control research community and has reached a high degree of maturity in its linear variant. Currently MPC research focuses on stochastic and nonlinear scenarios, robustness and fast optimisation or related computational aspects.

All algorithms to some extent form a trade-off between feasibility, optimality and inexpensive optimisation, but may not have systematic tools for doing this trade-off. The success of earlier industrial heuristic MPC algorithms motivated the research community to develop several algorithms with improved performance and feasibility. There are several successful theoretical approaches but few of them are exploited commercially for real-time implementation. One important issue for real-time implementation is the ability to do a systematic trade-off between feasibility, performance and computational burden when choosing from currently available algorithms. In recent years many authors have focussed on parametric solutions [5] or efficient implementations of online optimisers (e.g. for quadratic programming). However, this paper will follow a different route and consider the underlying structure of the MPC algorithm. Specifically, this paper proposes a systematic selection of an alternative parameterisation to enlarge the feasible region without too much detriment to performance (or optimality) and the computational burden.

In an MPC problem formulation, stability can be ensured using dual mode prediction by including a terminal constraint at the expense of the computational load [16]. In dual mode prediction, different formulations of the degrees of freedom (d.o.f.) or decision variables have been utilized using interpolation techniques [8] to enlarge feasible region without detriment to performance. There are different variants of interpolation, however, these methods are limited to small dimensional systems. Another approach in the literature is a concept of triple mode control [6], [7]; the triple mode strategy introduces an extra mode which may reduce the online computational burden with good performance and feasibility. However, in this strategy, the challenge is to find a suitable linear time varying (LTV) control law which enlarge feasible region with improved performance. One pragmatic solution to finding this law is tackled in [12] which used Laguerre parameterisations. However, this paper does not pursue Triple mode approaches as the offline complexity and decision making is substantially increased as compared to dual mode approaches.

More recently, alternative parameterisation based on Laguerre [17] and Kautz functions have been proposed to simplify the trade-off without increasing computational burden within dual mode prediction paradigm [9], [10], [11]. The main idea is to form the degrees of freedom in the predictions as a combination of either Laguerre or Kautz functions. These functions have proven to be a very effective for improving the volume of the feasible region with a limited number of d.o.f. These alternative parameterisations are generalised using orthonormal basis functions with Laguerre and Kautz functions as special cases [13].

The key question left to be resolved is, what is the best prediction dynamics to assume for predicted inputs, that is the d.o.f., to allow for a large feasible region without detriment to the closed loop performance and computational burden? In [15], different ways of parameterising the d.o.f. were investigated and a mechanism based on a Monte Carlo approach to define the best parameterisation using optimal sequences for numerous search directions was considered. However, a systematic choice for the underlying dynamics remains an open question. This paper focuses on systematic approaches to identify the best parameterisation dynamics using: (i) a multiobjective optimisation and (ii) a pragmatic choice.

This paper assumes that Laguerre, Kautz and Generalised

parameterisation are able to achieve large feasible regions while maintaining local optimality and a relatively low computational complexity [9], [10], [11], [12], [13], [14] and extends the earlier studies in [11], [13] in order to identify a systematic selection of the best parameterisation dynamics. Section II will give the necessary background about an optimal MPC, Laguerre MPC, generalised parameterisation for optimal MPC. Section III discusses two proposed schemes to identify the best parameterisation dynamics based on a multi-objective optimisation and a pragmatic approach. Numerical examples are presented in section IV and paper finishes with conclusions and future work in section V.

II. BACKGROUND

This section will summarise the background information related to nominal dual-mode MPC and the use of alternative parameterisations within MPC.

A. Problem formulation for MPC

Assume a discrete time linear time invariant (LTI) state space model of the form

$$x_{k+1} = Ax_k + Bu_k \quad (1)$$

where $x_k \in R^{n_x}$ and $u_k \in R^{n_u}$ which are the state vectors and the plant input respectively. Assume that the states and inputs at all time instants should fulfill the following constraints:

$$\underline{u} \leq u_k \leq \bar{u}; \quad \underline{\Delta u} \leq \Delta u_k \leq \overline{\Delta u}; \quad \underline{x} \leq x_k \leq \bar{x} \quad (2)$$

B. Nominal MPC algorithm

In dual-mode MPC it is assumed that one has total freedom in the choice of the input signal u_k up to horizon n_c subject to constraints. Beyond horizon n_c , a terminal control law with an asymptotic stabilising optimal feedback gain K is assumed. The ‘predicted’ control law [3], [16] takes the form

$$\begin{aligned} u_k &= -Kx_k + c_k, & k &= 0, \dots, n_c - 1, \\ u_k &= -Kx_k, & k &\geq n_c, \end{aligned} \quad (3)$$

where only first control move is ever implemented, c_k are degrees of freedom (d.o.f.) available for constrained handling. The terminal control law defines an invariant set χ_0 for state vector x_k [8]. This invariant set is also known as maximum admissible set (MAS) which satisfies all polytopic constraints with recursive use of the terminal control law $u_k = -Kx_k \in \chi_0$. The MAS is defined as:

$$\begin{aligned} \chi_0 &= \{x_0 \in R^{n_x} : \underline{x} \leq x_k \leq \bar{x}, \underline{u} \leq -Kx_k \leq \bar{u}, \\ & x(k+1) = Ax(k) + Bu(k), \forall k \geq 0\} \end{aligned} \quad (4)$$

In compact form defined as $\chi_0 = \{x_k : Mx_k \leq b\}$ for suitable M and b . Performance, either predicted or actual, will be assessed by the cost

$$J = \sum_{k=0}^{\infty} x_k^T Q x_k + u_k^T R u_k \quad (5)$$

Substituting the nominal model and predicted control values (3) into (5) and ignoring terms that do not depend upon the

d.o.f., one finds from [3] that the optimisation problem in (5) can be reformulated as:

$$\min_C J_c = C^T S C \text{ s.t. } Mx_k + NC \leq b; \quad (6)$$

where $C = [c_k^T, \dots, c_{k+n_c-1}^T]$. Details are in the literature [2], [3], [4]. The maximal control admissible set (MCAS) χ_c , the feasible set for optimal control problem in (6) that satisfies all polytopic constraints, is defined as:

$$\chi_c = \{x_k : \exists C, Mx_k + NC \leq b\} \quad (7)$$

The optimal MPC (OMPC) algorithm with guarantees of recursive feasibility and convergence, for the nominal case is given by solving QP optimisation (6) at every sampling instance then implementing the first component of C , that is c_k in the control law of (3). The algorithm is formulated as:

Algorithm 2.1: OMPC [16], [3]

$$c_k^* = \arg \min_{c_k} J_c \text{ s.t. } Mx_k + Nc_k \leq b;$$

Implement $u_k = -Kx_k + e_1^T c_k^*$ where $e_1^T = [1, 0, \dots, 0]$. When the initial states $x_k \in \chi_0$ then the optimising c_k^* is zero so the terminal control law $u_k = -Kx_k$ is implemented.

Remark 2.1: There is a well understood set of potentially conflicting objectives using OMPC e.g. between the desire for good performance and large feasible regions with the equally important desire to keep the number of d.o.f. small. In order for OMPC to obtain a large feasible region and good performance, a large number of d.o.f. or n_c is required.

C. Generalised parameterisation for Optimal MPC

Alternative parameterisation techniques for the d.o.f. in the future control values have been developed to improve the feasible region in nominal case. Laguerre, Kautz and generalised parameterisations have been proposed in [9], [11], [13] as an effective alternative to the standard basis set. This section will summarise Generalised optimal MPC.

1) *Generalised functions:* The generalised parameterisation [13] is defined using a higher order network such as:

$$\begin{aligned} \mathcal{G}_i(z) &= \mathcal{G}_{i-1}(z) \frac{(z^{-1} - a_1) \dots (z^{-1} - a_n)}{(1 - a_1 z^{-1}) \dots (1 - a_n z^{-1})}; \\ 0 &\leq a_k < 1, \quad k = 1 \dots n \end{aligned} \quad (8)$$

With $\mathcal{G}_1(z) = \frac{\sqrt{(1-a_1^2) \dots (1-a_n^2)}}{(1-a_1 z^{-1}) \dots (1-a_n z^{-1})}$. The generalised function with $a_k, \forall k = 1, \dots, n$ gives [13]

$$\begin{aligned} \text{Laguerre :} & \quad \mathcal{G}_i = \mathcal{L}_i, \quad \text{if } a_k = [a] \\ \text{Kautz :} & \quad \mathcal{G}_i = \mathcal{K}_i, \quad \text{if } a_k = [a, b]. \end{aligned} \quad (9)$$

The Laguerre function is a special case of a generalised function and may be computed using the following state-space

dynamics model:

$$\mathcal{G}(k+1) = \underbrace{\begin{pmatrix} a & 0 & 0 & 0 & \dots \\ \beta & a & 0 & 0 & \dots \\ -a\beta & \beta & a & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}}_{A_G} \mathcal{G}(k); \quad (10)$$

$$\mathcal{G}(0) = \sqrt{1-a^2}[1, -a, a^2, \dots]^T; \quad \beta = 1-a^2$$

2) *GOMPC: generalised functions and MPC*: The predictions using d.o.f. based on generalised functions [13] are:

$$C = \begin{pmatrix} c_k \\ \vdots \\ c_{k+n_c-1} \\ \vdots \end{pmatrix} = \begin{pmatrix} \mathcal{G}(0)^T \\ \mathcal{G}(1)^T \\ \vdots \end{pmatrix} \rho = H_G \rho \quad (11)$$

where ρ is the n_G dimension decision variable when one uses the first n_G column of H_G . The only difference between Laguerre MPC and Kautz or generalised MPC is that of H_G matrix. For further details readers are referred to [11] and [13].

Algorithm 2.2: GOMPC

$$\begin{aligned} \rho^* &= \arg \min_{\rho} \rho^T \left[\sum_{i=0}^{\infty} A_G^i \mathcal{G}(0) S \mathcal{G}(0)^T (A_G^i)^T \right] \rho \\ \text{s.t. } & Mx_k + NH_G \rho \leq b \end{aligned} \quad (12)$$

Define $c_k^* = H_G \rho_k^*$ and implement $u_k = -Kx_k + e_i^T c_k^*$, where e_i^T is the i -th standard basis vector.

Remark 2.2: It is straightforward to show, with conventional arguments, that all algorithms (i.e. LOMPC, KOMPC, GOMPC) using terminal constraints within MPC problem formulation provides recursive feasibility and Lyapunov stability.

III. BEST ALTERNATIVE PARAMETERISATION SELECTION

In generalised parameterisation there are two clear choices within the future input predictions. First one can choose the order of the dynamics, that is the number of poles a_i in $G_i(z)$, and second is the parameter selection(s), that is the actual values of a_i . This section discusses the systematic selection of the parameterisation dynamics by asking what impact this choice has on feasibility and performance?

A. The best choice for order of the prediction dynamics

The prediction dynamics for the 3rd order prediction dynamics from (8) (which can easily be extended to n^{th} order) can be defined as:

$$\mathcal{G}(k+1) = \underbrace{\begin{pmatrix} b & 0 & 0 & \dots \\ b & c & 0 & \dots \\ -ab & (1-ac) & a & \dots \\ ab^2 & -b(1-ac) & (1-ac) & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}}_{A_G} \mathcal{G}(k) \quad (13)$$

$$\begin{aligned} \mathcal{G}(0) &= \gamma[1, 1, 1, -a, ab, -abc, a^2bc, \dots]^T, \\ \gamma &= \sqrt{(1-a^2)(1-b^2)(1-c^2)}, \end{aligned}$$

The dynamic structure of the prediction in (13) is quite generic with distinct eigenvalues. To fulfill the algebraic relations in (11), A_G used in (13) must be a square matrix with a maximum number of dimensions the same as n_c . Moreover a key observation from the prediction matrix structure is that $\dim(A_G) = n_c$ is an upper bound on dynamic dimensions. In fact, one could select $\dim(A_G) \leq n_c$.

B. Multiobjective solution to select best parameterisation

The main objective of this paper is to formulate a systematic method for handling the compromise between feasibility, performance and computational burden. This section shows how one can produce trade-off curves between feasible volume v , performance loss β and computational burden (implicitly linked to n_c) of the parameterised MPC problem. Such trade-off curves allow us to formulate a multi-objective optimisation problem as a function of parameterisation parameters $\alpha = [a_1, \dots, a_m] \in (0, 1)^m$:

$$\begin{aligned} J(n_c, n) &:= \min_{\alpha} \beta, \max_{\alpha} v \\ \text{s.t. } & Mx + NH\rho \leq b, \\ & v = \frac{\text{vol}(P_H)}{\text{vol}(P_{opt})}, \\ \beta &= \frac{1}{n} \sum \frac{J_H(x) - J_{opt}(x)}{J_{opt}(x)}, \\ \alpha &= [a_1, \dots, a_m], \quad 0 < a_i < 1, \quad m \leq n_c, \\ & \mathcal{G}(k+1) = A_G \mathcal{G}(k), \\ & H = (\mathcal{G}^T(0) \quad \mathcal{G}^T(1) \quad \dots)^T, \end{aligned} \quad (14)$$

where $P_{opt} := \{(x, c) | Mx + Nc \leq b\}$ represents the MCAS with $n_c = 20$ (used as an approximate for the global maximum MCAS) for comparison, $P_H = \{(x, \rho) | Mx + NH\rho \leq b\}$ is the polytope sliced by the matrix H , $\text{vol}(\cdot)$ is the volume, $J_{opt}(x) = \text{avg.} \{J_c(x, c) | (x, c) \in P_{opt}\}$ and $J_H(x) = \text{avg.} \{J_c(x, H\rho) | (x, \rho) \in P_H\}$ for convex function $J_c(x, c)$.

1) *Volume approximation*: Computing the volume of a high dimensional polytope is a complex task, and can, in the worst case be exponential in the size of the data M and N or NH . Consequently, this paper approximates the volume. First select a large number of equi-spaced (by solid angle) or random directions in the state space i.e. $x = (x_1, \dots, x_n)$ and then, for each direction, the distance from the origin to the boundary of MCAS is determined by solving a linear programming (LP) and clearly the larger the distance, hereafter denoted as radius, the better the feasibility. Finally radii are normalised against the radii obtained for P_{opt} . Although this might be considered somewhat arbitrary, it seems a pragmatic way of indicating relative volumes of different feasible regions compared to the best feasible shape with a large n_c . One might argue that a precise volume measurement would, in some sense, be an equally arbitrary comparison.

2) *Performance approximation*: Similarly in case of calculating the performance loss, it is converted from integral to

sampled sum i.e.

$$\beta \approx \frac{1}{n} \sum \frac{J_H(x)}{J_{opt}(x)} - 1 \quad (15)$$

That is, one computes the predicted performance for all the given state directions and compares each of these to the 'global' optimum. Equation (15) normalises these so that the best achievable performance would correspond to a β measure of unity.

By deploying explicit numeric measures of volume and performance, the multi-objective optimisation problem is able to generate trade-off curves between the d.o.f. n_c , average radii and average performance loss of the parameterised problem with different parameterisation settings.

C. Pragmatic selection

In practice it is known from parameterisation insight that the optimal ρ_k is highly nonlinear in terms of its dependence upon the current state x_k ; more over the solution is to some extent unpredictable. The following gives a simple approach to identify a pragmatic selection of pole location(s) and order selection of parameterisation dynamics.

1) *Order selection*: The generalised functions with higher order orthonormal functions have more flexibility to improve feasibility while retaining good performance with a limited number of d.o.f. [13]. So $\dim(A_G) = n_c$ is a sub-optimal choice to select order parameterisation dynamics.

2) *Parameter selection*: It is observed from simulation results that there is a relation between the closed loop poles and good locations of parameterisation dynamics. The pole locations of generalised functions can be selected to be equal to poles of the closed loop system using an optimal gain K ; this may be sub-optimal but is efficient.

D. Summary

In summary, there are two choices: (i) the order of parameterisation dynamics ($\dim(A_G)$) and (ii) parameter value(s) or pole location(s) ($[a_1, \dots, a_n]$). One can use multi-objective optimisation to find $\dim(A_G)$ and $[a_1, \dots, a_n]$. Another key observation is that the maximum $\dim(A_G) \leq n_c$ and closed loop eigenvalues or pole locations with an optimal gain K are a pragmatic choice to tune the parameterisation dynamics.

IV. NUMERICAL EXAMPLES

In this section three numerical examples are presented to illustrate the efficacy of the proposed approach. The aim is to compare two aspects: (i) feasible regions; (ii) performance loss for the generalised parameterisation dynamics with different parameter(s) or pole location(s) selection by varying the d.o.f. or n_c . The trade-off curves are shown between average radius and performance loss as a function of the different parameterisation parameter(s) and d.o.f. or n_c . The parallel coordinates are also shown to represent an alternative variation of the parameter(s) and the effects on average performance loss and average feasibility gain. The multi-objective optimisation is done using the NSG-II Matlab toolbox.

A. Examples

Three numerical examples are simulated using weightings $Q = C^T C$, $R = 2$ and are simulated by varying $n_c = [2, \dots, 6]$ and with different parameter values. Trade-off curves between average radius and average performance loss are plotted by solving the multi-objective optimisation with varying parameter values. Similarly the effect on average performance loss and average radius are shown by varying n_c . Conversely, Table 1 shows the average performance loss and average radius using a pragmatic approach to parameter selection.

1) Example 1:

$$A = [1.2]; B = [1]; C = [1]; \\ -0.2 \leq u_k \leq 0.2; -0.2 \leq \Delta u_k \leq 0.2; -5 \leq Cx_k \leq 5;$$

2) Example 2:

$$A = \begin{bmatrix} 0.6 & -0.4 \\ 1 & 1.4 \end{bmatrix}; B = \begin{bmatrix} 0.2 \\ 0.05 \end{bmatrix}; C = [1 \quad -2.2] \\ -0.8 \leq u_k \leq 1.5; -0.4 \leq \Delta u_k \leq 0.4; -5 \leq Cx_k \leq 5;$$

3) Example 3:

$$A = \begin{bmatrix} 1 & 0.1 & 0.3 \\ 0 & 1 & 0.1 \\ 0.4 & 0 & 1 \end{bmatrix}; B = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}; \\ C = [1 \quad 0 \quad 0] \\ -2 \leq u_k \leq 2; -1 \leq \Delta u_k \leq 1; -5 \leq Cx_k \leq 5;$$

B. Optimal selection based on multiobjective solution

The multi-objective procedure was run for different sample directions, which were chosen uniformly on the unit circle. The parameters n_c representing d.o.f. were chosen in the range $\{2, \dots, 6\}$. The results for Examples 1, 2, 3 are shown in Figures 1-2, 3-5 and 6-7 respectively. In all plots, colour circles represent the parameter values α .

Figures 1, 3 and 6 show the parallel coordinates and trade-off between average performance loss and average radius by varying n_c with different parameter values α . Parallel coordinates are used to plot all axes representing different objectives parallel to each other. Only the Laguerre function parameterisation simulation results are shown in the figures, but there are similar results for higher order parameterisations i.e. Kautz and 3rd order functions. These trade-off curves may be used as a selection criteria to choose best function parameterisation α .

Figures 2, 4 and 7 show the trade-off curves between average performance loss and average radius for $n_c = 4$ with different parameterisation dynamics. It is observed that 3rd order function parameterisation improves both the feasible region without too much detriment to performance. An average performance loss and feasible region gain for example 2 is shown in figure 5 by varying $n_c \in \{2, \dots, 6\}$ for a Laguerre parameterisation. One can see that best parameter selection may improve the feasible region and performance with a limited number of d.o.f. n_c .

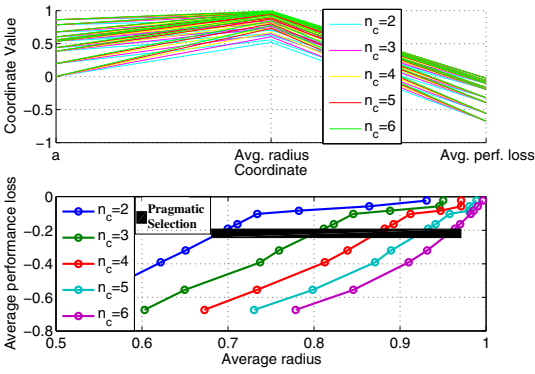


Fig. 1. Parallel coordinates and trade-off curves for Example 1

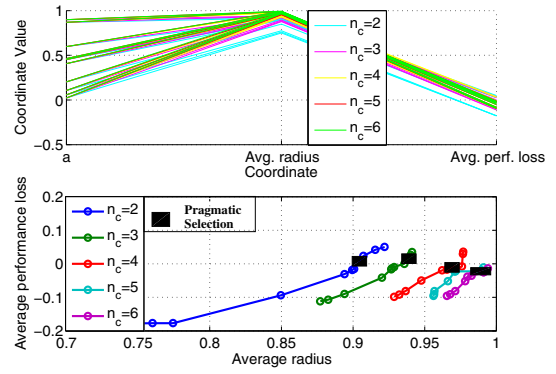


Fig. 3. Parallel coordinates and trade-off curves for Example 2

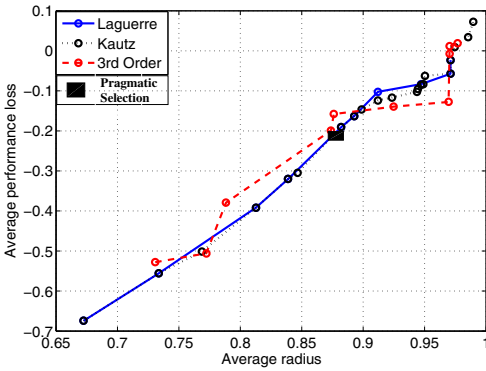


Fig. 2. Trade-off as a function of the parameter α : $n_c = 4$ for Example 1

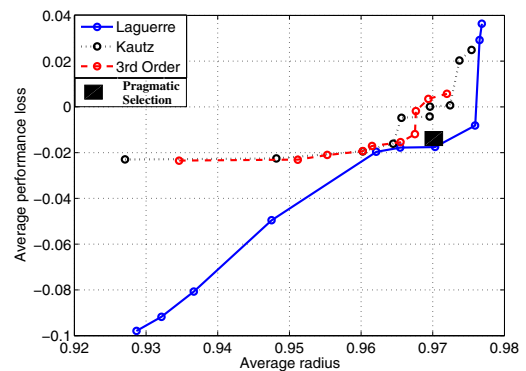


Fig. 4. Trade-off as a function of the parameter α : $n_c = 4$ for Example 2

C. Sub optimal selection based on closed loop eigenvalues

Table I, II and III show the pragmatic selection of parameterisation dynamics using closed loop eigenvalues with an optimal gain K . It is clear from these that pragmatic choices are sub optimal. The selection of parameterisation is based on closed loop eigenvalues so parameterisation order is selected based on example dimensions.

Laguerre parameterisation is the only choice for example 1. With a sub optimal choice of $n_c = 2$ achieves 70 % of feasible set and 22 % of performance loss. Similarly in example 2, there are two parameterisation choices Laguerre and Kautz functions. The Laguerre function achieved 90 % of feasible set with 6 % performance loss. On the otherhand the Kautz function achieved 95 % of feasible set with 1 % performance loss with $n_c = 2$. In example 3, there are three parameterisation choices i.e. Laguerre, Kautz and 3rd order functions. All achieved more than 90 % global feasible set and maximum of 8 % of performance loss.

Remark 4.1: It is clear from figures 1-4, 6 and 7 that the pragmatic choice has solution in vicinity of an optimal pareto solution.

Remark 4.2: The optimisation structure of an approximate global optimal control is different to the parameterised one, so in few cases performance may even improve.

V. CONCLUSION

The main contribution of this paper was to present a systematic approach to selecting the best alternative parameterisation for MPC. Two techniques were discussed based on a multi-objective optimisation and a pragmatic approach using closed loop poles. Examples demonstrate that in many cases nominal closed loop poles are a good sub optimal choice to tune the parameterisation dynamics. However an optimal selection may be made using a multi-objective optimisation technique. In future work there is a need to further investigate in parallel issues such as: will these choices lead to an efficient online optimisation QP structure and/or low complexity multi-parametric solution? Finally, it is an interesting future area to consider these choices within a robust scenario.

TABLE I
PRAGMATIC SELECTION FOR α

Example 1			
Parameterisation	n_c	Avg. perf. loss	Avg. radius
Laguerre a=0.52	2	22%	0.6904
	3	22%	0.8011
	4	22%	0.8720
	5	22%	0.9237
	6	22%	0.9578

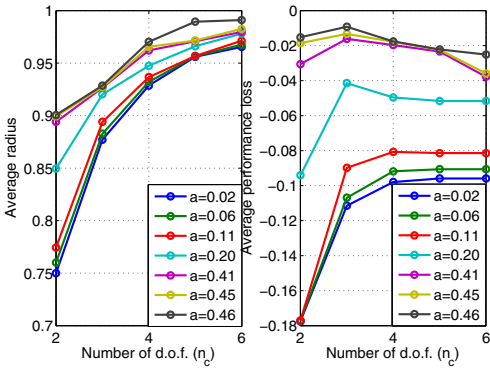


Fig. 5. Average radius and average performance loss for Example 2

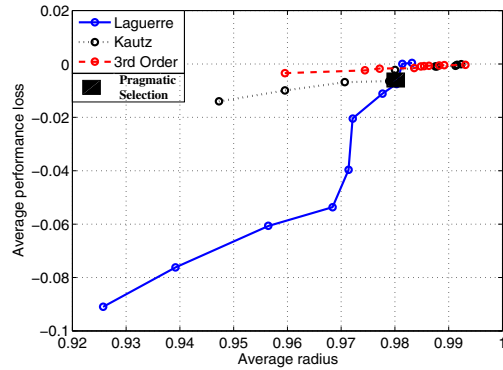


Fig. 7. Trade-off as a function of the parameter α : $n_c = 4$ for Example 3

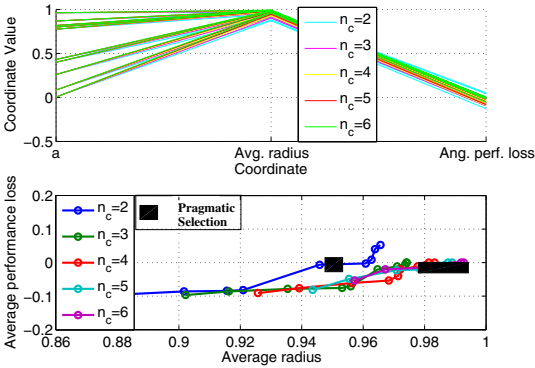


Fig. 6. Parallel coordinates and trade-off curves for Example 3

TABLE III
PRAGMATIC SELECTION FOR α

Example 3			
Parameterisation	n_c	Avg. perf. loss	Avg. radius
Laguerre $a=0.36$	2	0.2 %	0.9113
	3	0.2%	0.9494
	4	0.2%	0.9658
	5	0.2%	0.9790
	6	0.2%	0.9873
$a=0.74$	2	8 %	0.9580
	3	6 %	0.9731
	4	2 %	0.9865
	5	0.7 %	0.9906
	6	0.6 %	0.9936
Kautz $(a,b)=0.74 \pm 0.02j$	2	8 %	0.9580
	3	8 %	0.9600
	4	0.4%	0.9764
	5	0.4%	0.9831
	6	0.4%	0.9954
Generalised $(a,b,c)=0.36,0.74 \pm 0.02j$	2	2 %	0.9113
	3	0 %	0.9494
	4	0 %	0.9689
	5	0 %	0.9930
	6	0 %	0.9934

REFERENCES

[1] E. Camacho and C. Bordons, *Model predictive control*, Springer, 2003.
 [2] D. Q. Mayne, J. B. Rawlings, C. Rao and P. Scokaret, "Constrained model predictive control", *Automatica*, vol. 36, 2000, pp 789-814.
 [3] J. A. Rossiter, *Model-based predictive control, A practical approach*, CRC Press, London; 2003.
 [4] E. Gilbert and K. Tan, "Linear systems with state and control constraints: The theory and application of maximal output admissible set", *IEEE Trans. on Automatic Control*, vol. 36, 1991, pp 1008-1020.
 [5] P. Grieder, Efficient computation of feedback controllers for constrained systems, PhD Thesis, ETH, 2004
 [6] L. Imsland and J. A. Rossiter, "Time varying terminal control", *Proc. 16th IFAC World Congress*, Prague, Czech Republic, 2005.
 [7] L. Imsland, J. A. Rossiter, B. Pluymers and J. Suykens, "Robust triple mode MPC", *IJC*, vol. 81, 2005, pp 679-689.

TABLE II
PRAGMATIC SELECTION FOR α

Example 2			
Parameterisation	n_c	Avg. perf. loss	Avg. radius
Laguerre $a=0.72$	2	6 %	0.9098
	3	3 %	0.9548
	4	2%	0.9683
	5	2 %	0.9799
	6	2 %	0.9896
Kautz $(a,b)=0.72 \pm 0.37j$	2	1 %	0.9468
	3	0 %	0.9584
	4	0.5 %	0.9829
	5	2 %	0.9885
	6	2 %	0.9907

[8] J. A. Rossiter, B. Kouvaritakis, M. Bacic, "Interpolation based computationally efficient predictive control", *IJC*, vol. 77, 2004, pp 290-301.
 [9] J. A. Rossiter and L. Wang, "Exploiting Laguerre functions to improve the feasibility/performance compromise in MPC", *Proc. CDC.*, Cancun, Mexico, 2008.
 [10] J. A. Rossiter, L. Wang and G. Valencia-Palomo, "Efficient algorithms for trading off feasibility and performance in predictive control", *IJC*, vol. 83, 2010, pp 789-797.
 [11] B. Khan, J. A. Rossiter and G. Valencia-Palomo, "Exploiting Kautz functions to improve feasibility in MPC", *Proc. 18th IFAC World Congress*, Milan, Italy, 2011.
 [12] B. Khan and J. A. Rossiter, "Triple Mode MPC or Laguerre MPC: a comparison", *Proc. ACC*, San Francisco, California, USA, 2011.
 [13] B. Khan and J. A. Rossiter, "Generalised Parameterisation for MPC", *Proc. The 13th IASTED Inter. Conf. On Intelligent Systems and Control (ISC)*, Cambridge, UK, 2011.
 [14] B. Khan and J. A. Rossiter, "Computational Efficiency of Laguerre MPC using Active set Methods", *Proc. The 13th IASTED Inter. Conf. On Intelligent Systems and Control (ISC)*, Cambridge, UK, 2011.
 [15] G. Valencia-Palomo, J. A. Rossiter, C. N. Colin, R. Gondhalekar and B. Khan, "Alternative parameterisations for predictive control: how and why?", *Proc. ACC*, San Francisco, USA, 2011.
 [16] P. Scokaert and J. B. Rawlings, "Constrained linear quadratic regulation", *IEEE Trans. on Automatic Control*, vol. 43, 1998, pp 1163-1168.
 [17] L. Wang, *Model predictive control design and implementation using MATLAB*, Springer, 2009.