OPTIMAL CONTROL APPLICATIONS AND METHODS
Optim. Control Appl. Meth. 2010; **00**:1–15
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/oca

Long horizon input parameterisations to enlarge the region of attraction of MPC

B. Khan[1]*, G. Valencia-Palomo[2], J. A. Rossiter[3], C. N. Jones[4] and R. Gondhalekar[5]

[1]*Department of Electrical Engineering COMSATS Institute of Information Technology, Abbottabad, Pakistan,* [2]*Instituto Tecnológico de Hermosillo, Av. Technológico y Periférico Poniente, México,* [3]*Department of Automatic Control and Systems Engineering, University of Sheffield, UK,* [4]*Automatic Control Laboratory, Department of Mechanical Engineering, École Polytechnique Fédérale De Lausanne, Switzerland,* [5]*Department of Chemical Engineering, University of California, Santa Barbara, USA*

SUMMARY

In this paper the efficacy of structured and unstructured parameterisations of the degree of freedom within a predictive control algorithm is investigated. While several earlier papers investigated the enlargement of the region of attraction using structured prediction dynamics, little consideration has been given to the potential of unstructured parameterisations to handle the trade-off between the region of attraction, performance and computational burden. This paper demonstrates how unstructured dynamics can be both selected and used effectively and furthermore gives a comparison with structured methods. Copyright © 2010 John Wiley & Sons, Ltd.

Received ...

KEY WORDS: Alternative parameterisation; MPC; Region of attraction; Performance; Computational burden.

1. INTRODUCTION

Model Predictive control (MPC) [13, 17, 3] is popular because it handles multivariable processes with constraints in a systematic fashion, but to achieve this, the online implementation may require a challenging optimisation. In this paper, there is a well understood set of conflicting objectives, e.g., between the desire for good performance and large regions of attraction, with the equally important desire to keep the number of degrees of freedom (d.o.f.) small in order to maintain an implementable computational complexity.

Several authors have looked at this issue, some well known ones being focussed on multi-parametric solutions [1], fast optimisations [23], time varying control laws [12], interpolation between two different control strategies, [15, 19, 18], and blocking [2, 7]; the latter two methods form foundation concepts for parametric methods proposed in [20] where the key development is that the effective horizon range of the d.o.f. (for constraint handling) is far greater than the number of d.o.f. (this is not the case for conventional algorithms such as Generalised Predictive Control (GPC)).

*Correspondence to: Department of Electrical Enigneering COMSATS Institute of Information Technology, Abbottabad, Pakistan.

In the earlier work of [9, 11] ellipsoidal approximations of the constraints were implicitly embedded within the MPC problem, resulting in a formulation that is equivalent to that of [20]. However, ellipses are ill-suited to approximating constraints that are polyhedral and/or asymmetric. Furthermore, when using ellipsoidal approaches [4] there is no further gain in the size of the ellipsoidal region of attraction when the order of the parameterisation dynamic exceeds the system dimension.

In light of [4] an obvious extension to the parameterisations proposed in [20] that are based on Laguerre polynomials was to consider a wider class of functions; for reasons of numerical conditioning the focus was given to generalised orthonormal functions, such as Kautz [10]. Specifically, it was shown that in many cases changing the parameterisation allowed substantial improvements in the volume of the region of attraction with little or no detriment to performance. However this gives rise to several questions:

1. Is there an alternative to structured ‘generalised function dynamics’ for parameterising the input trajectories which would yield further benefits? For example multi-parametric approaches have shown that the constrained optimal control law is piecewise affine in the state. Furthermore, blocking [2] is another example of an alternative structure.
2. The algorithms of [20] are based on implicit assumption that the choice of initial condition for the parameterised dynamics is unimportant, as this was typical in the original works [22]; is this true?

This paper is organised as follows: after presenting the background in Section 2, Section 3 is considered the impact of the initial condition for the dynamic embedded in the degrees of freedom and thus whether there are preferred choices for this. Section 4 is proposed an alternative mechanism for exploiting the desired input prediction space in an efficient manner but which is not necessarily linked to a fixed dynamic; this will be termed an unstructured parameterisation. Section 5 analyses unstructured parameterisations in detail and formulates a convex optimisation problem for performing a strategic tradeoff between cost performance and volumes of the sets of feasible states. Section 6 will present numerical examples followed by conclusions in Section 7.

2. BACKGROUND

2.1. *Modelling and a standard optimal MPC algorithm*

Assume a standard state-space model of the form:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k; \quad \mathbf{y}_k = \mathbf{C}\mathbf{x}_k; \tag{1}$$

with $\mathbf{x}_k \in \mathbb{R}^{n_x}$, $\mathbf{y}_k \in \mathbb{R}^{n_y}$ and $\mathbf{u}_k \in \mathbb{R}^{n_u}$ which are the state vector, the measured output and the plant input, respectively.

The performance index to be minimised (with respect to $\mathbf{u}_k, \mathbf{u}_{k+1}, \ldots$) is

$$\begin{array}{ll} J & = \sum_{i=0}^{\infty} (\mathbf{x}_{k+i+1})^T \mathbf{Q} (\mathbf{x}_{k+i+1}) + (\mathbf{u}_{k+i})^T \mathbf{R} (\mathbf{u}_{k+i}) \\ s.t. & \left\{ \begin{array}{l} (1), \underline{\mathbf{u}} \leq \mathbf{u}_k \leq \overline{\mathbf{u}}, \ \ \underline{\mathbf{y}} \leq \mathbf{y}_k \leq \overline{\mathbf{y}} \ \ \forall k \geq 0, \\ \mathbf{u}_{k+i} = -\mathbf{K}\mathbf{x}_{k+i} \ \ \forall i \geq n_c \end{array} \right. \end{array} \tag{2}$$

with $\mathbf{Q}$, $\mathbf{R}$ positive definite state and input cost weighting matrices. $\mathbf{K}$ is the optimal feedback gain minimising J in the absence of constraints.

Practical limitations imply that only a finite number, that is n_c, of free control moves can be used [21]. For these cases, (2) is implemented [14] by imposing that the state $\mathbf{x}_{n_c}$ must be contained in a polytopic control invariant set (that is the Maximum Admissible Set (MAS)): $\mathcal{X}_{MAS} = \{x \in \mathbb{R}^{n_x} \mid C\phi^i x \in \mathbb{Y}, -K\phi^i x \in \mathbb{U}, \forall i \geq 0\}$, where $\phi = A - BK$, $\mathbb{Y} = \{\mathbf{y} \in \mathbb{R}^{n_y} \mid \underline{\mathbf{y}} \leq \mathbf{y} \leq \overline{\mathbf{y}}\}$ and $\mathbb{U} = \{\mathbf{u} \in \mathbb{R}^{n_u} \mid \underline{\mathbf{u}} \leq \mathbf{u} \leq \overline{\mathbf{u}}\}$. For simplicity of notation, the MAS can also be described in the form $\mathcal{X}_{MAS} = \{\mathbf{x} \in \mathbb{R}^{n_x} \mid \mathbf{M}_0\mathbf{x} \leq \mathbf{d}_0\}$ for appropriate $\mathbf{M}_0, \mathbf{d}_0$.

DOI: 10.1002/oca

The degrees of freedom can be reformulated in terms of a new variable $\mathbf{c}_k$ using dual mode paradigm for guaranteeing nominal stability [17] i.e.,

$$\mathbf{u}_{k+i} = \begin{cases} -\mathbf{K}\mathbf{x}_{k+i} + \mathbf{c}_{k+i}; & i = 0, ..., n_c - 1; \\ -\mathbf{K}\mathbf{x}_{k+i}; & i \geq n_c; \end{cases} \tag{3}$$

and hence the equivalent optimisation to (2) is

$$\min_{\underset{\rightarrow}{\mathbf{c}}} \quad \underset{\rightarrow}{\mathbf{c}}^T \mathbf{S} \, \underset{\rightarrow}{\mathbf{c}} \quad s.t. \;\; \mathbf{M}\mathbf{x} + \mathbf{N} \, \underset{\rightarrow}{\mathbf{c}} \leq \mathbf{d}; \tag{4}$$

with $\underset{\rightarrow}{\mathbf{c}} = [\mathbf{c}_0^T, \ldots, \mathbf{c}_{n_c-1}^T]^T$. Details of how to compute positive definite matrix $\mathbf{S}$, matrices $\mathbf{N}$, $\mathbf{M}$ and vector $\mathbf{d}$ may be found in the following literature: [6, 14, 17].

Definition 2.1
Let $\mathcal{X}_{MCAS}$ be the set of initial states $\mathbf{x}$ for which the optimal control problem (4) is feasible (often denoted as the feasible region):

$$\mathcal{X}_{MCAS} = \{\mathbf{x} \in \mathbb{R}^{n_x} \mid \exists \, \underset{\rightarrow}{\mathbf{c}} \in \mathbb{R}^{n_c n_u}, \mathbf{M}\mathbf{x} + \mathbf{N} \, \underset{\rightarrow}{\mathbf{c}} \leq \mathbf{d}\}.$$

A typical conflict in MPC design is between the desire for a large volume of $\mathcal{X}_{MCAS}$, which increases with n_c, and the desire for n_c to be small as this links to the computational load of the optimisation. It has been noted that the formulation of (3) is inefficient in that typically a large n_c is required for large volumes [20].

2.2. *Optimisation of prediction dynamics*

In [4], a convex formulation of the optimisation of prediction dynamics is proposed to enlarge the region of attraction using as optimally tuned a terminal control law as is possible in combination with any other stabilising law. Specifically, the predicted input trajectories can be generated by incorporation of a dynamic feedback law:

$$\mathbf{u}_k = -\mathbf{K}\mathbf{x}_k + \mathbf{C}_c\mathbf{A}_c\mathbf{c}_k; \quad \mathbf{c}_k = \mathbf{A}_c\mathbf{c}_{k-1} \tag{5}$$

where $\mathbf{C}_c$ and $\mathbf{A}_c$ are variables that are used to optimise the size of the associated feasible invariant ellipsoid [4]. It was also shown that there is no further gain in volume of the region of an ellipsoidal region of attraction when the prediction horizon exceeds the system dimension. However, the proposed optimisation for selecting this feedback is based on ellipsoids and hence will typically be conservative in volume.

The design parameter in this optimisation are matrices, $\mathbf{C}_c$ and $\mathbf{A}_c$. It is clear that this concept is equivalent to the parameterisations in [10, 20] with the minor difference that the latter considered a special case of $\mathbf{A}_c$, being a lower triangular matrix of a particular structure and $\mathbf{C}_c$ depends on the parameter of $\mathbf{A}_c$. The main advantage of parameetrisations in [10] is that the orthonormal functions may define $\mathbf{A}_c$ intuitively (without optimisation) and moreover is well defined for orders less than the system state dimension and asymmetric constraints; the algorithm of [4] does not handle such cases.

2.3. *GOMPC: Generalised functions and MPC*

Generalised Optimal MPC (GOMPC) is a dual-mode MPC algorithm [20, 10] where the input predictions are parameterised in terms of generalised functions using an input prediction akin to (5). The generalised functions can be computed using an arbitrary order discrete transfer function network in z-transform (typically we use order $n \in \{1, 2, 3\}$):

$$g_{ii}(z) = g_{ii-1}(z) \frac{(z^{-1} - a_1)^{ii-1} \ldots (z^{-1} - a_n)^{ii-1}}{(1 - a_1 z^{-1})^{ii} \ldots (1 - a_n z^{-1})^{ii}}; \qquad 0 \leq a_j < 1, \quad j = 1, \ldots, n, \tag{6}$$

with $g_1(z) = \frac{\sqrt{(1-a_1^2)\dots(1-a_n^2)}}{(1-a_1 z^{-1})\dots(1-a_n z^{-1})}$. For example, in case of $n = 4$ a state space formulation is given by

$$\underbrace{\begin{bmatrix} g_{1,k+1} \\ g_{2,k+1} \\ g_{3,k+1} \\ g_{4,k+1} \\ g_{5,k+1} \\ \vdots \end{bmatrix}}_{\mathbf{G}_{k+1}} = \underbrace{\begin{bmatrix} a_2 & 0 & 0 & 0 & 0 \\ a_2 & a_3 & 0 & 0 & 0 \\ a_2 & a_3 & a_4 & 0 & 0 \\ -a_1 a_2 & -a_1 a_3 & 1 - a_1 a_4 & a_1 & 0 \\ a_1 a_2^2 & a_1 a_2 a_3 & -a_2(1 - a_1 a_4) & 1 - a_1 a_2 & a_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}}_{\mathbf{A}_G} \underbrace{\begin{bmatrix} g_{1,k} \\ g_{2,k} \\ g_{3,k} \\ g_{4,k} \\ g_{5,k}, \\ \vdots \end{bmatrix}}_{\mathbf{G}_k}; \tag{7}$$

$$\mathbf{G}_0 = \sqrt{(1-a_1^2)\dots(1-a_4^2)}\,[1,\ 1,\ 1,\ -a_1,\ a_1 a_2, \dots]^T.$$

Input predictions are defined using (3) and:

$$\mathbf{c}_k = \mathbf{G}_k^T \eta_k, \quad \mathbf{G}_{k+1} = \mathbf{A}_G \mathbf{G}_k, \quad \forall k. \tag{8}$$

Consequently predictions for $\mathbf{c}_k$ evolve over an infinite horizon via the following dynamic relationship:

$$\underrightarrow{\mathbf{c}}_k = H_G \underrightarrow{\eta}_k, \quad H_G = [\mathbf{G}_0^T, \dots, \mathbf{G}_{n_c-1}^T, \dots]. \tag{9}$$

The associated feasible region is given as

$$\mathcal{X}_G = \{\mathbf{x}_k \in \mathbb{R}^{n_x} \mid \exists \underrightarrow{\eta}_k \in \mathbb{R}^{n_c n_u}, \mathbf{M}_G \mathbf{x}_k + \mathbf{N}_G \mathbf{H}_G \underrightarrow{\eta}_k \le \mathbf{d}_G\}. \tag{10}$$

for suitably defined $\mathbf{N}_G, \mathbf{H}_G, \mathbf{M}_G, \mathbf{d}_G$.

Algorithm 2.1
The GOMPC Algorithm is summarised as:

$$\underrightarrow{\eta}_k^* = \arg\min_{\underrightarrow{\eta}_k} \ \underrightarrow{\eta}_k^T \mathbf{H}_G^T S \mathbf{H}_G \underrightarrow{\eta}_k \ \ s.t.\ \ \mathbf{M}_G \mathbf{x}_k + \mathbf{N}_G \mathbf{H}_G \underrightarrow{\eta}_k \le \mathbf{d}_G; \tag{11}$$

Define $\underrightarrow{\mathbf{c}}_k^* = \mathbf{H}_G \underrightarrow{\eta}_k^*$ and implement $\mathbf{u}_k = -\mathbf{K}\mathbf{x}_k + \mathbf{e}_1^T \underrightarrow{\mathbf{c}}_k^*$, $\mathbf{e}_1^T = [I_{n_u}, 0, \dots, 0]$.

Remark 2.1
Readers will note that $\mathbf{G}_0$ is needed to define the initial condition.

Remark 2.2
Systematic mechanisms to choose the best parameterisation dynamics were discussed in [10]. If GOMPC uses $a_j = 0$, then $\mathbf{G}_0 = [1, 0, \dots]$ and GOMPC is equivalent to Optimal MPC (OMPC).

3. THE BEST CHOICE FOR INITIAL CONDITION IN THE PREDICTION DYNAMICS

The background section has shown two clear choices for GOMPC within the future input predictions. First, one can choose the implied dynamic $\mathbf{A}_G$ and, second, one can choose the initial condition $\mathbf{G}_0$. This section explores the second of these choices by asking what impact the choice $\mathbf{G}_0$ has on the region of attraction (10)? It is clear from (8) that the choice has a clear link to the input predictions and thus the impact needs investigation. A key question is whether the initialization of $\mathbf{G}_0$ can be exploited to improve either the size of the region of attraction, or closed loop performance?

Lemma 3.1
The predictions $\mathbf{c}_{k+i,k}$[‡] can be considered as the output of a linear time invariant state-space model with initial condition linked to $\mathbf{G}_0$.

[‡] $k + i, k$ means the prediction for sample $k + i$ made at sample k.

Proof
Define an arbitrary state-space model with dynamic matrix Φ^T as follows

$$\mathbf{w}_{k+1} = \Phi^T \mathbf{w}_k; \quad \mathbf{c}_k = \underset{\rightarrow}{\eta}^T \mathbf{w}_k; \quad \mathbf{w}_0 = \mathbf{G}_0. \tag{12}$$

Forming predictions for this gives $\mathbf{c}_k = \underset{\rightarrow}{\eta}^T \mathbf{G}_0, \ \mathbf{c}_{k+1} = \underset{\rightarrow}{\eta}^T \Phi^T \mathbf{G}_0, \ \mathbf{c}_{k+2} = \underset{\rightarrow}{\eta}^T (\Phi^T)^2 \mathbf{G}_0, \ \ldots$ □

Lemma 3.2
If Φ has distinct eigenvalues, then there exists an alternative state-space model with the same input/output relationship but different initial condition and different output matrix that gives the same output predictions as (12).

Proof
In place of initial condition $\mathbf{G}_0$, instead use $\mathbf{Z}_0$ so now define the model.

$$\mathbf{v}_{k+1} = \Phi^T \mathbf{v}_k; \quad \mathbf{c}_k = \underset{\rightarrow}{\tilde{\eta}}^T \mathbf{v}_k; \quad \mathbf{v}_0 = \mathbf{Z}_0. \tag{13}$$

The requirement is that the output predictions of models (12) and (13) given next can be made the same but with $\mathbf{Z}_0 \neq \mathbf{G}_0$ and $\underset{\rightarrow}{\eta} \neq \underset{\rightarrow}{\tilde{\eta}}$. Hence, prove

$$\mathbf{c}_{k+n} = \underset{\rightarrow}{\eta}^T (\Phi^T)^n \mathbf{G}_0 = \underset{\rightarrow}{\tilde{\eta}}^T (\Phi^T)^n \mathbf{Z}_0, \ \forall n. \tag{14}$$

First, decompose Φ using its eigenvalue/vector decomposition into the form $\Phi^T = \mathbf{V}\Lambda\mathbf{V}^{-1}$ and substitute in (14):

$$\mathbf{c}_{k+n} = \underset{\rightarrow}{\eta}^T \mathbf{V}\Lambda^n \mathbf{V}^{-1} \mathbf{G}_0 = \underset{\rightarrow}{\tilde{\eta}}^T \mathbf{V}\Lambda^n V^{-1} \mathbf{Z}_0. \tag{15}$$

From this it is clear that

$$\mathbf{c}_{k+i} = \sum_j \lambda_j^i \alpha_j; \quad \left\{ \begin{array}{c} \alpha_j = [\underset{\rightarrow}{\eta}^T \mathbf{V}]_j [\mathbf{V}^{-1}\mathbf{G}_0]_j; \\ \text{or} \\ \alpha_j = [\tilde{\eta}^T \mathbf{V}]_j [\mathbf{V}^{-1}\mathbf{Z}_0]_j. \end{array} \right. \tag{16}$$

Consequently, there always exists a choice of $\underset{\rightarrow}{\tilde{\eta}}$ so that the output of model (13) replicates (12), as long as the initial condition $\mathbf{Z}_0$ is not orthogonal to any eigenvector of Φ. □

However, the reader will notice that the implied state-space matrix with the generalised functions has repeated eigenvalues when the order of dynamics is less than the number of d.o.f. (i.e. $n < n_c$, where n is number of poles) and thus does not have a simple decomposition.

Lemma 3.3
For dynamic matrices of the form given in (7) with repeated eigenvalues, the initial condition still has no bearing on the reachable space of future $\mathbf{c}_{k+i}$.

Proof
This follows similar lines to Lemma 3.2 with the only difference that a non-simple Jordan form is required for n_c is greater than the number of poles of generalised function dynamics. It is still possible to decompose the matrix as $\Phi^T = \mathbf{V}\mathbf{J}_d\mathbf{V}^{-1}$ where $\mathbf{J}_d$ is no longer strictly diagonal but all powers of $\mathbf{J}_d$ are upper triangular, and consequently the same proof as previously applies, but with slightly more complex algebra.

$$\mathbf{c}_{k+n} = \underset{\rightarrow}{\eta}^T \mathbf{V}\mathbf{J}_d^n \mathbf{V}^{-1} \mathbf{G}_0 = \underset{\rightarrow}{\tilde{\eta}}^T \mathbf{V}\mathbf{J}_d^n \mathbf{V}^{-1} \mathbf{Z}_0. \tag{17}$$

Similarly as in Lemma 3.2

$$\mathbf{c}_{k+i} = \sum_j (\lambda_j^i + (i-1)\lambda_j^{i-1})\alpha_j + \beta_j; \quad \left\{ \begin{array}{c} \alpha_j = [\underrightarrow{\eta}^T \mathbf{V}]_j [\mathbf{V}^{-1}\mathbf{G}_0]_j; \\ \beta_j = [\underrightarrow{\eta}^T \mathbf{W}\mathbf{G}_0]_j; \\ \text{or} \\ \alpha_j = [\tilde{\eta}^T \mathbf{V}]_j [\mathbf{V}^{-1}\mathbf{Z}_0]_j; \\ \beta_j = [\underrightarrow{\eta}^T \mathbf{W}\mathbf{Z}_0]_j; \end{array} \right. \tag{18}$$

where **W** depends upon Jordan blocks. Consequently, there always exists a choice of $\underrightarrow{\tilde{\eta}}$ so that the output of model (13) replicates (12), as long as the initial condition $\mathbf{Z}_0$ is not orthogonal to any eigenvector of Φ. □

In summary, subject to some mild conditions on including components of all the eigenvectors, the choice of $\mathbf{G}_0$ has no impact on the reachable space of $\mathbf{c}_{k+i}$ and thus is not a parameter that needs to be considered further.

4. PARAMETERISATIONS USING UNSTRUCTURED PREDICTIONS

One could argue that conventional algorithms such as GPC and OMPC have unstructured input predictions in that there is no explicit or implicit link between the different values. GOMPC has structured predictions because there is an explicit link through the model of (8). However, the advantage of GOMPC is that the effective horizon of the input perturbations $\mathbf{c}_k$ is infinite, whereas for OMPC the effective horizon is just n_c!, in order to maintain an implementable computational complexity. Nevertheless, such a fixed structure is restrictive and indeed the insights of multi-parametric solutions [1] make this very clear.

A question that follows from these observations is the following: Does there exist an *unstructured* perturbation class for $\underrightarrow{\mathbf{c}}$ that has a large effective horizon but which can be captured with a low number of d.o.f., and, would such a class bring any advantages? In simple terms this could be interpreted as allowing a totally open choice for $\mathbf{H}_G$ (see (9)).

4.1. *The reachable space for the input predictions*

The flexibility within the input predictions (8) is given from $\underrightarrow{\mathbf{c}}_k = \mathbf{H}_G \underrightarrow{\eta}_k$. The row dimension of $\mathbf{H}_G$ dictates how far into the future one wishes to use non-zero $\mathbf{c}_k$ and the column dimension dictates the number of d.o.f.. While for OMPC $\mathbf{H}_G$ is square, here the assumption is that it is tall and thin, or more precisely $n_{c2} \times n_c$ (blocks in case of more than one input), where $n_{c2} > n_c$. The problem is then, how does one determine the optimum columns for $\mathbf{H}_G$? These columns define the flexibility in the input predictions.

4.2. *Monte Carlo approaches to find the search directions*

It is known that the mapping from current state $\mathbf{x}_k$ to optimal offset vector $\underrightarrow{\mathbf{c}}_k$ is nonlinear; consequently a range of different search directions within feasible sets are needed to capture the required flexibility in $\underrightarrow{\mathbf{c}}_k$ for large feasible region. Here, a simple Monte Carlo approach is taken to capture those search directions that have 'most value'.

Algorithm 4.1
Search directions

1. Choose a large value of $n_c = n_{c2}$ for OMPC consistent with finding a large enough feasible region.
2. Define equi-spaced points on the surface of the unit n_x dimensional sphere centered at the origin. Stretch these directions to the outer boundary of the MCAS (for given large $n_c = n_{c2}$) and denote them as v_i.

3. For each point v_i, determine the optimal $\underset{\rightarrow}{\mathbf{c}}_k$ and denote as $\underset{\rightarrow}{\mathbf{c}}_{vi}$.
4. Form a matrix $\mathbf{P} = [\underset{\rightarrow}{\mathbf{c}}_{v1}, \underset{\rightarrow}{\mathbf{c}}_{v2}, \cdots, \underset{\rightarrow}{\mathbf{c}}_{vm}]$.
5. Find the singular value decomposition of $\mathbf{P}$ as $\mathbf{P} = \mathbf{X}\Sigma\mathbf{Y}^*$.
6. Define $\mathbf{H}_F = \mathbf{X}(:, 1 : n_c)$ where $n_c = n_\eta$ is now taken to be the desired number of d.o.f.. It is assumed that the first n_c columns of $\mathbf{X}$ correspond to the largest singular values.

It is clear that in some objective sense, this choice of $\mathbf{H}_F$ captures the best finite number of search directions, on average, to capture the optimal $\underset{\rightarrow}{\mathbf{c}}_k$ required on the boundary of the MCAS. Moreover, one can inspect the singular values of $\mathbf{P}$ in Σ to determine what might be the best number of columns to take.

4.3. *Feasible OMPC*

This section shows briefly how the $\mathbf{H}_F$ matrix of the previous subsection is used to define a Feasible OMPC (FOMPC) algorithm.

Algorithm 4.2
FOMPC

1. Select the maximum number n_{c2} of non-zero $\mathbf{c}_{k+i}$ terms in the predictions and the number of d.o.f. n_c to be used online; use Algorithm 4.1 to determine $\mathbf{H}_F$. Let $\underset{\rightarrow}{\mathbf{c}}_k = \mathbf{H}_F \underset{\rightarrow}{\eta}_k$.
2. Define $J_F = \underset{\rightarrow}{\eta}_k^T \mathbf{S}_F \underset{\rightarrow}{\eta}_k$, where $\mathbf{S}_F = \mathbf{H}_F^T \mathrm{diag}\{\mathbf{S}, \ldots, \mathbf{S}\}\mathbf{H}_F$.
3. Find the optimum $\underset{\rightarrow}{\eta}_k^*$ from:

$$\underset{\rightarrow}{\eta}_k^* = \arg\min_{\underset{\rightarrow}{\eta}} \quad J_F \quad s.t. \;\; \mathbf{M}\mathbf{x}_k + \mathbf{N}\mathbf{H}_F \underset{\rightarrow}{\eta}_k \le \mathbf{d}.$$

4. Define $\underset{\rightarrow}{\mathbf{c}}_k^* = \mathbf{H}_F \underset{\rightarrow}{\eta}_k^*$ and implement the control law $\mathbf{u}_k = -\mathbf{K}\mathbf{x}_k + \mathbf{e}_1^T \underset{\rightarrow}{\mathbf{c}}_k^*$.

4.4. *Recursive feasibility and convergence*

Here, by not prescribing any specific structure to $\mathbf{H}_F$, the specific structure of (7) is lost and also the literature standard recursive feasibility/convergence result. A simple procedure does exist in the literature to recover this guarantee, but at the cost of introducing one additional d.o.f. [15]. In simple terms, one appends the d.o.f. in $\mathbf{H}_F \underset{\rightarrow}{\eta}_k$ with one additional direction, that is the tail of the optimised $\underset{\rightarrow}{\mathbf{c}}_k$ from the previous sample. This is not discussed further, but interesting reader is referred to [15].

5. CONVEX OPTIMISATION OF FOMPC FOR MAXIMUM VOLUME PROJECTIONS

As the examples will show FOMPC generates a large feasible set for a low number of d.o.f.. In this section, whether FOMPC also provides a good trade-off between the volume of the resulting region of attraction and cost performance is investigated. The goal of this section is to produce a pareto surface between the d.o.f. n_η, the resulting volume of the feasible set (as a function of the average radius α) and the level of suboptimality β of FOMPC.

Our goal is to solve the following optimisation problem as a function of $(\alpha, \beta) \in [0, 1]^2$:

$$\begin{aligned} \tilde{J}(\alpha, \beta) = & \min_{\mathbf{H}_F} \quad n_\eta \\ s.t. \quad & \mathbf{d} \le \alpha \mathbf{M} x_i + \mathbf{N} u_i, \; \forall i = \{1, \ldots, n\}, \\ & \beta \ge \frac{1}{n} \sum_{i=1}^{n} \frac{J_F(\alpha x_i) - J_{opt}(\alpha x_i)}{J_{opt}(\alpha x_i)}, \\ & u_i = \mathbf{H}_F \eta_i, \quad \mathbf{H}_F \in \mathbb{R}^{m \times n_c}. \end{aligned} \tag{19}$$

Computing the volume of, or the integral over, a polytope is complex, and can, in the worst of the case be exponential in the size of **M** and **N**. Here, the volume is approximated by the average radius α of the MCAS determined by choosing a number of points $x = \{x_1, \ldots, x_n\}$ equi-spaced on the unit hyper-sphere centered at origin and solving a series of linear programming (LP) in order to move the points to the boundary. The predicted performance, for given points x_i are represented by the optimised values of closed loop $J_{opt}(x_i), J_F(x_i)$ for global OMPC and FOMPC.

Constraint $u_i = \mathbf{H}_F \eta_i$ is bi-linear in the decision variable $\mathbf{H}_F$ and η_i and therefore the optimisation problem above (19) is non-convex optimisation problem. We propose in this section a convex relaxation of (19) based on the nuclear norm.

The bilinear constraint is equivalent to

$$u_i \in \text{Range}(\mathbf{H}_F), \quad \mathbf{H}_F \in \mathbb{R}^{m \times n_\eta}, \quad \forall i = 0, \ldots, n. \tag{20}$$

The only property of interest of the matrix $\mathbf{H}_F$ is its number of columns (i.e., the rank, because $\mathbf{H}_F$ is tall) and so we can replace the constraint $u_i = \mathbf{H}_F \eta_i$ with the following rank-condition

$$\text{rank}(\mathbf{U}) \leq n_\eta \tag{21}$$

where $\text{U} = \begin{bmatrix} u_0, \ldots, u_n \end{bmatrix}$. Minimising the rank of a matrix $\mathbf{U} \in \mathbb{R}^{m \times n}$ is a non-convex problem and is in general NP-hard.

The nuclear norm is a convex heuristic for rank minimisation that was proposed in [5] and shown in [16] to be the convex envelope, or the closest convex function to the rank operation

$$||\mathbf{U}||_\star = \sum_{i=1}^{m} \sigma_i(\mathbf{U}) \tag{22}$$

where $\sigma_i(\mathbf{U})$ is the i^{th} singular value of **U**. In last few years, minimisation of the l_1 norm has been used as a convex approximation of cardinality minimisation, or to promote sparsity in the decision vector of optimisation problems. Since the singular values of a matrix are all positive, the nuclear norm of **U** is equal to the l_1 norm of the vector formed from the singular values of **U**. As a result, minimising nuclear norm in (22) leads to sparsity in the vector of singular values, or equality to a low-rank matrix **U**.

We now relax the optimisation to its convex envelope and the optimisation can be re-cast as the following semi-definite program (SDP) [5].

$$\begin{aligned} \hat{J}(\alpha, \beta) = \min_{\mathbf{U}_\star} \quad & tr(\mathbf{V}_1) + tr(\mathbf{V}_2) \\ s.t. \quad & \begin{bmatrix} \mathbf{V}_1 & \mathbf{U}^T \\ \mathbf{U} & \mathbf{V}_2 \end{bmatrix} \geq 0 \\ & \mathbf{d} \geq \alpha \mathbf{M} x_i + \mathbf{N} u_i; \\ & \beta \geq \frac{1}{n} \sum_{i=1}^{n} \frac{J_F(\alpha x_i) - J_{opt}(\alpha x_i)}{J_{opt}(\alpha x_i)} \end{aligned} \tag{23}$$

where we introduce the symmetric matrices $\mathbf{V}_1 \in \mathbb{R}^{n \times n}$ and $\mathbf{V}_2 \in \mathbb{R}^{m \times m}$ as decision variables.

Let $\mathbf{U}^\star$ be the optimal solution of (23) and $n_\eta^\star$ be the rank of $\mathbf{U}^\star$. A matrix $\mathbf{H}_F \in \mathbb{R}^{m \times n_\eta^\star}$ that span the range of $\mathbf{U}^\star$ can then be found via a singular decomposition of $\mathbf{U}^\star$. We term the resulting procedure of solving SDP (23) and then computing the optimal $\mathbf{U}^\star$ as optimised FOMPC.

6. NUMERICAL EXAMPLES

This section gives some numerical illustrations to compare the efficacy of OMPC, GOMPC and FOMPC. Specifically, the focus is on the comparison of the volume of the MCAS against the number of optimisation variables, that is, it considers to what extent FOMPC uses a more systematic parameterisation of the d.o.f. within the predictions to allow for maximal gains in the size of the regions of attractions with small numbers of d.o.f.

For numerical simulations, a large number of points equi-spaced (by solid angle for 2-dimensional systems) or random selection or chosen uniformly on the unit hyper-sphere; feasible volumes are inferred from the distance, here denoted radii, to the feasibility boundary in each direction. A notional or pragmatic maximum (of cost performance and size of the regions of attractions) for comparison is taken as that obtained from OMPC with $n_c = 20$ (theoretical maxima can vary [8]).

The following models/constraints are used.

Example 1

$$\mathbf{A} = \begin{bmatrix} 0.6 & -0.4 \\ 1 & 1.4 \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} 0.2 \\ 0.05 \end{bmatrix}; \quad \mathbf{C} = \begin{bmatrix} 1 & -2.2 \end{bmatrix};$$

$$-1.5 \leq \mathbf{u}_k \leq 0.8; \ |\Delta \mathbf{u}_k| \leq 0.4; \ |\mathbf{x}_{i,k}| \leq 5; \ \mathbf{Q} = \mathbf{I}; \ \mathbf{R} = 2.$$

Example 2

$$\mathbf{A} = \begin{bmatrix} 0.9146 & 0 & 0.0405 & 0.1 \\ 0.1665 & 0.1353 & 0.0058 & -0.2 \\ 0 & 0 & 0.1353 & 0.5 \\ -0.2 & 0 & 0 & 0.8 \end{bmatrix};$$

$$\mathbf{B} = \begin{bmatrix} 0.054 & -0.075 \\ 0.005 & 0.0147 \\ 0.864 & 0 \\ 0.5 & 0.2 \end{bmatrix}; \mathbf{C} = \begin{bmatrix} 1.799 & 13.21 & 0 & 0.1 \\ 0.823 & 0 & 0 & -0.3 \end{bmatrix};$$

$$|\Delta \mathbf{u}_{i,k}| \leq 2; \ |\mathbf{u}_k| \leq \begin{bmatrix} 1 \\ 2 \end{bmatrix}; \ |\mathbf{x}_{i,k}| \leq 10; \ \mathbf{Q} = \mathbf{C}^T\mathbf{C}; \ \mathbf{R} = \mathbf{I}.$$

Readers should note that for example 2, the number of d.o.f. is $n_c n_c$, and that in this case $n_u = 2$. The parameterisation dynamics for GOMPC are selected using a multi-objective optimisation [10].

This section contains two main comparisons:

1. The achievable feasible volumes with FOMPC and GOMPC[§].
2. A comparison between a default FOMPC and one arising using the Nuclear norm search assesseing the performance and feasibility trade-off.

6.1. *Comparison of the size of regions of attractions for OMPC, GOMPC and FOMPC*

The feasible regions for Example 1 are shown in Figure 1. The average radii for examples 1, 2 are shown in figures 3, 4 respectively. It is clear that FOMPC has a larger MCAS than GOMPC for the same number of d.o.f. Indeed Figure 3 shows that, for this example, FOMPC gets to within 85% of the global MCAS with just 3 d.o.f. whereas, GOMPC require 7 d.o.f. and OMPC requires 11 d.o.f. For example 2, FOMPC needs just 2 d.o.f.!.

Consideration of the singular values for the respective $\mathbf{H}_L, \mathbf{H}_F$ matrices shown in Figure 2 shows that a choice of $n_\eta \in \{3, 4\}$ captures all the key directionality, whereas for GOMPC, many more columns are needed.

6.2. *Comparison between FOMPC and optimised FOMPC*

This section demonstrates that while the default FOMPC gives the largest feasible volumes, for a given n_η, use of optimisation (23) enables a compromise between feasibility and performance. For both examples, the optimisation (23) was run for each $\alpha \in \{0.01, 0.025, 0.5, 0.75, 0.99\}$ and a fixed value of $\beta = 5\%$.

[§] Other work has already demonstrated that GOMPC outperforms both OMPC and the algorithm of [4].

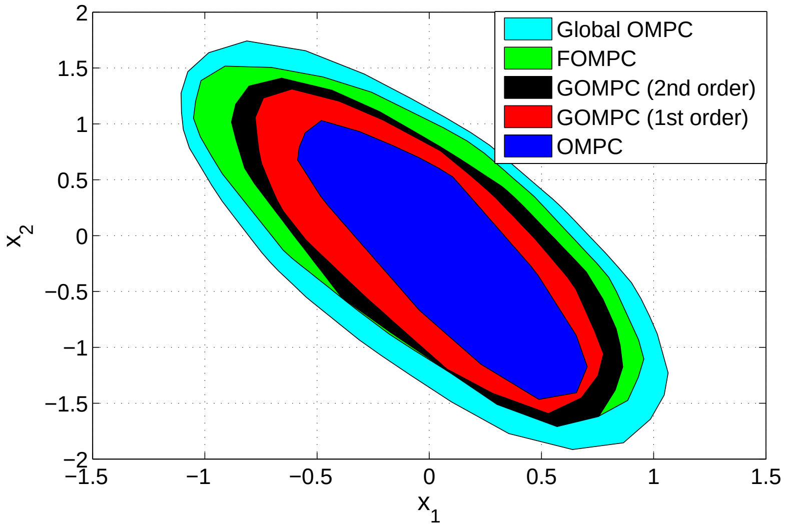

Figure 1. Comparison of MCAS for $n_c = n_\eta = 2$ for Example 1.

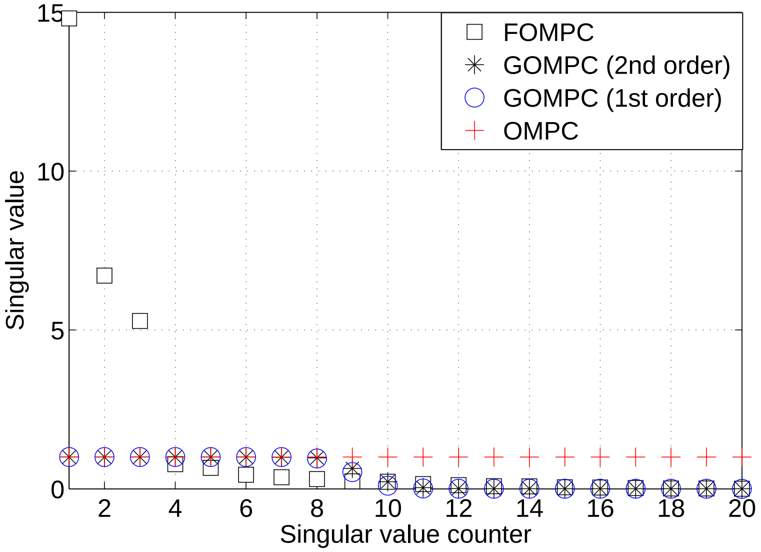

Figure 2. Comparison of singular values of $\mathbf{H}_L$ for Example 1.

1. For Example 1 for each α resulted in matrices $\mathbf{H}_F$ with $\{1, 1, 3, 6, 9\}$ columns respectively. Figures 5, 6 show the resulting average radii of the regions of attraction and the performance drop resulting from each considered parameterisation. Figure 9 shows the trade-offs between radius and performance loss as a function of the d.o.f.
2. For Example 2, the performance loss is significantly less for all cases than was seen for Example 1. For each α resulted in matrices $\mathbf{H}_F$ with $\{2, 4, 5, 4, 5\}$ columns respectively. The results are shown in Figures 7, 8, and 10.

DOI: 10.1002/oca

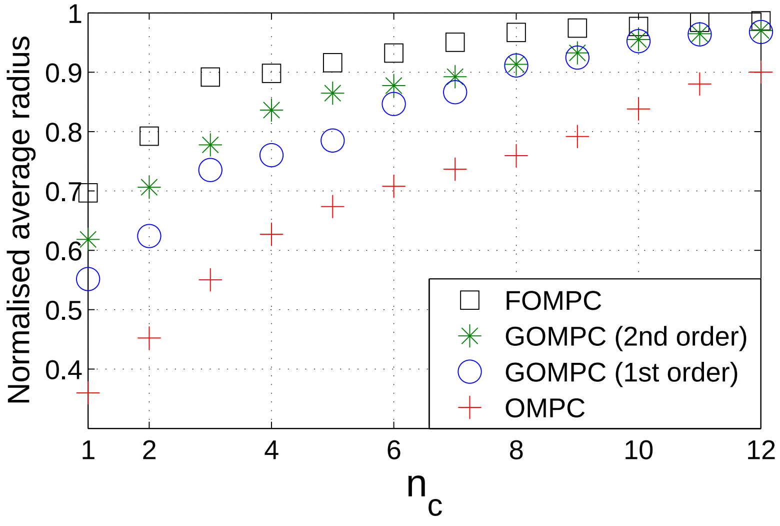

Figure 3. Comparison of average MCAS radii as n_c, n_η vary for Example 1.

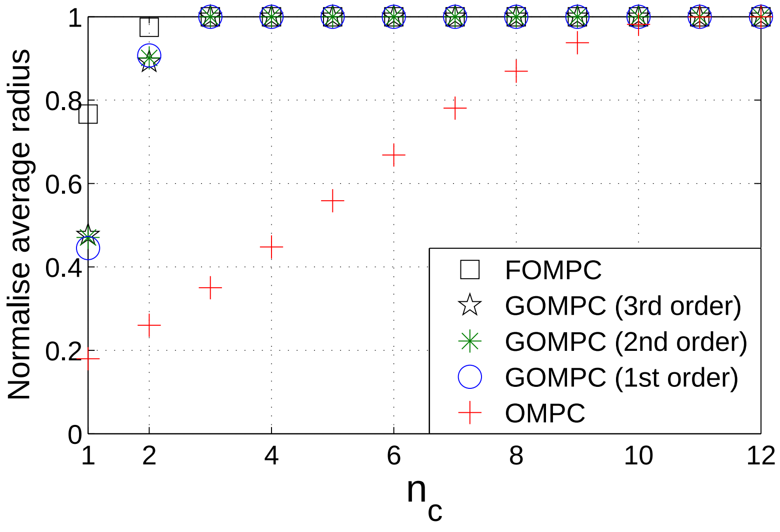

Figure 4. Comparison of average MCAS radii as n_c, n_η vary for Example 2.

7. CONCLUSIONS

In this paper a novel mechanism for parameterising the d.o.f. within a dual-mode MPC algorithm was proposed. The proposal builds on the insights of parameterised methods, which have demonstrated how to achieve *long* effective input horizons using small numbers of d.o.f.. Here, the concept is radically altered by identifying the optimal subset of long horizon 'unstructured' perturbations which are then applied to the input predictions. It is shown that using well defined perturbations allows substantial improvements in the volume of the set of feasible states, for the same numbers of d.o.f.. However, given that feasibility and performance requirements are often in conflict, the paper also proposes systematic tools for analysing this trade-off and thus allowing a design choice.

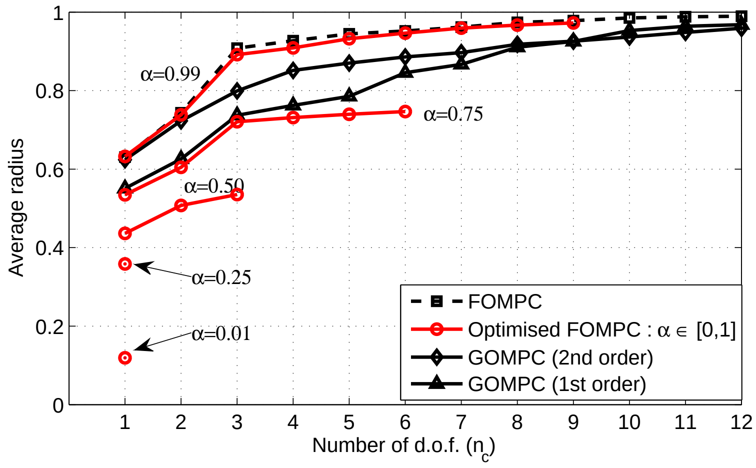

Figure 5. Average radius as a function of α for Example 1.

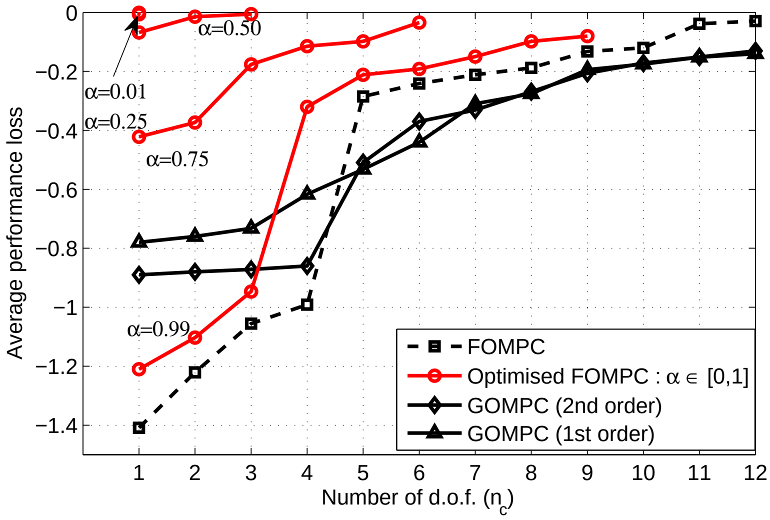

Figure 6. Average performance loss as a function of α for Example 1.

Future work will look at several questions: (i) How well do these methods extend to large dimensional systems or indeed to make use of system or direction specific information? (ii) How would the approach be modified to deal with the uncertain case? (iii) While the proposal reduces the required number of d.o.f., there is a need to investigate the potential for developing efficient online MPC optimisations.

REFERENCES

1. A. Bemporad, M. Morari, V. Dua, and E.N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, 2002.
2. R. Cagienard, P. Grieder, E.C. Kerrigan, and M. Morari. Move blocking strategies in receding horizon control. *Journal of Process Control*, 17(6):563–570, 2007.
3. E.F. Camacho and C. Bordons. *Model predictive control*. Springer, 2004.

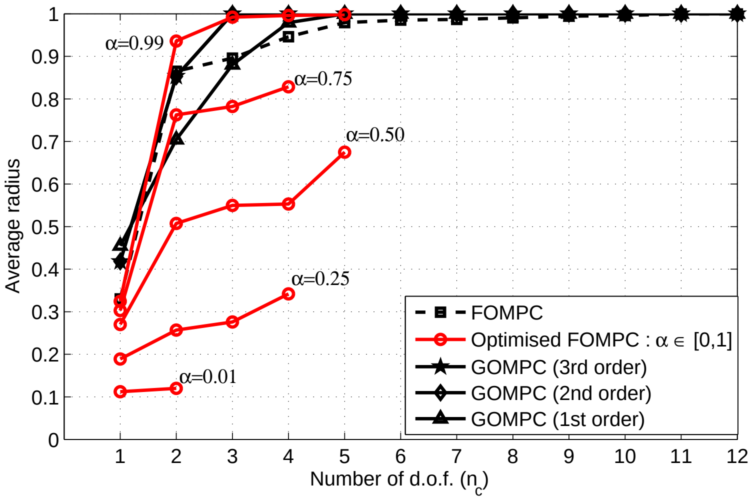

Figure 7. Average radius as a function of α for Example 2.

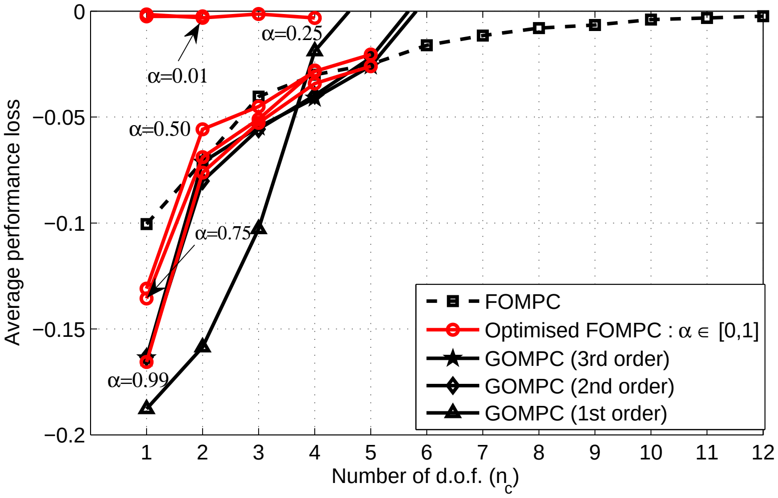

Figure 8. Average performance loss as a function of α for Example 2.

4. M. Cannon and B. Kouvaritakis. Optimizing prediction dynamics for robust MPC. *IEEE Trans Aut Control*, 50(11):1892–1897, 2005.
5. M. Fazel, H. Hindi, and S. Boyd. A rank minimization heuristic with application to minimum order system approximation. In *American Control Conference*, New York, USA, 2001.
6. E.G. Gilbert and K.T. Tan. Linear systems with state and control constraints: The theory and application of maximal output admissible sets. *IEEE Trans Aut Control*, 36(9):1008–1020, 1991.
7. R. Gondhalekar and J. Imura. Least-restrictive move-blocking model predictive control. *Automatica*, 46(7):1234–1240, 2010.
8. P. Grieder, F. Borrelli, F. Torrisi, and M. Morari. Computation of the constrained infinite time linear quadratic regulator. *Automatica*, 40(4):701–708, 2004.
9. L. Imsland, N. Bar, and B. Foss. More efficient predictive control. *Automatica*, 41(8):1395–1403, 2005.
10. B. Khan and J. A. Rossiter. Alternative parameterisation within predictive control: a systematic selection. *International Journal of Control*, 86(8):1397–1409, 2013.
11. B. Kouvaritakis, J.A. Rossiter, and J. Schuurmans. Efficient robust predictive control. *IEEE Trans Aut Control*, 45(8):1545–1549, 2000.
12. D. Limon, T. Alamo, and EF Camacho. Enlarging the domain of attraction of mpc controllers. *Automatica*, 41(4):629–635, 2005.

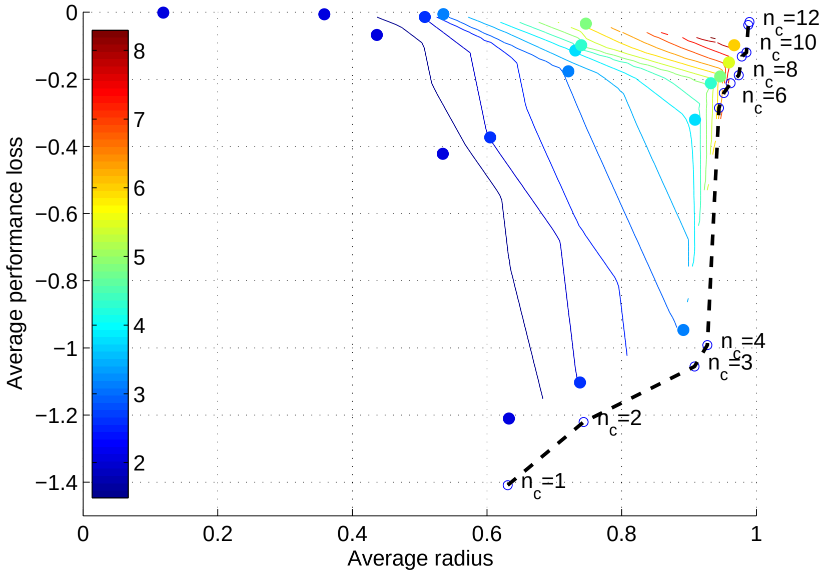

Figure 9. Trade-off between radius and performance loss for Example 1. Dashed black line: FOMPC; Colored dots: Optimised FOMPC; Colored-line: level-sets of constant d.o.f.; Color-bar: d.o.f. (r).

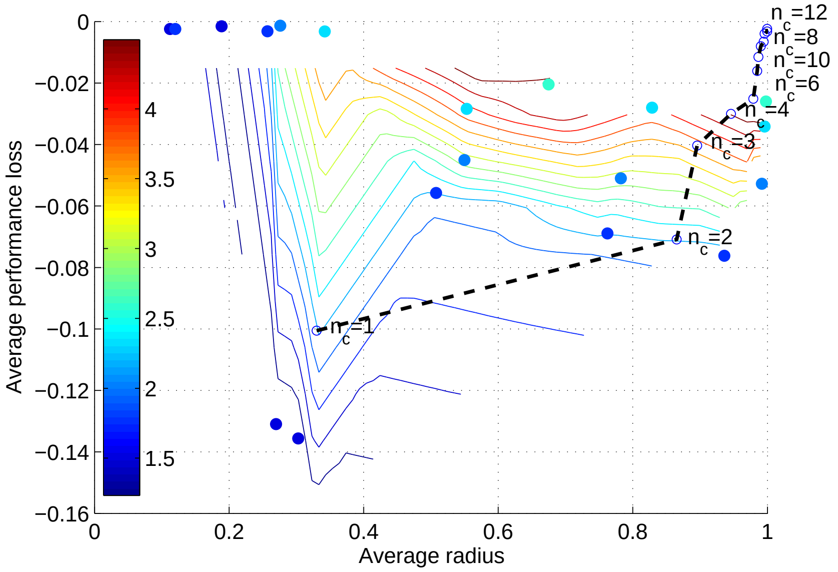

Figure 10. Trade-off between radius and performance loss for Example 2. Dashed black line: FOMPC; Colored dots: Optimised FOMPC; Colored-line: level-sets of constant d.o.f.; Color-bar: d.o.f. (r).

13. J.M. Maciejowski. *Predictive control with constraints*. Prentice Hall, 2002.
14. D.Q. Mayne, J.B. Rawlings, C.V. Rao, and P.O. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, 2000.
15. J.A. Mendez, B. Kouvaritakis, and J.A. Rossiter. State-space approach to interpolation in MPC. *International Journal of Robust and Nonlinear Control*, 10(1):27–38, 2000.
16. B. Recht, M. Fazel, and P. A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimizatoion. *SIAM review*, 52(3):471–501, 2010.

17. J.A. Rossiter. *Model-based predictive control, a practical approach*. CRC Press, London, 2003.
18. J.A. Rossiter and Y. Ding. Interpolation methods in model predictive control: an overview. *International Journal of Control*, 83(2):297–312, 2010.
19. J.A. Rossiter and P. Grieder. Using interpolation to improve efficiency of multiparametric predictive control. *Automatica*, 41(4):637–643, 2005.
20. J.A. Rossiter, L. Wang, and G. Valencia-Palomo. Efficient algorithms for trading off feasibility and performance in predictive control. *International Journal of Control*, 83(4):789–797, 2010.
21. P.O.M. Scokaert and J.B. Rawlings. Constrained linear quadratic regulation. *IEEE Trans Aut Control*, 43(8):1163–1169, 1998.
22. L. Wang. Discrete model predictive controller design using Laguerre functions. *Journal of Process Control*, 14(2):131–142, 2004.
23. Y. Wang and S. Boyd. Fast model predictive control using online optimization. *IEEE Transactions on Control Systems Technology*, 18(2):267–278, 2010.