Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Compromises between feasibility and performance within linear MPC

Yihang Ding, * **J.A Rossiter** **

* *Automatic Control and System Engineering, University of Sheffield, UK (e-mail: cop04yd@sheffield.ac.uk)*
** *(e-mail: j.a.rossiter@sheffield.ac.uk)*

Abstract: This paper explores the issues of feasibility and performance within predictive control. Conventional thinking is that there is typically a trade off between performance and the volume of the feasible region. However, this paper seeks to show that the trade off is often not as stark as might be expected and in fact one can sometimes gain huge amounts in feasibility with an almost negligible loss in performance while using a simple and conventional MPC algorithm.

Keywords: Constraints, Feasibility, Performance, Computational Efficiency, Contours

1. INTRODUCTION

Techniques for linear Model Predictive Control (MPC) Camacho et *al* (2005); Rossiter (2003) are now fairly well understood and widely applied with great success, especially in the large chemical and petro-chemical industries. For cases where the underlying system is open-loop stable and there are only input constraints, it is also obvious using recent insights that a DMC Cutler et *al* (1980) or GPC Clarke et *al* (1987) type of algorithm will give reasonable performance for almost any input horizon, so long as the output horizon is longer than the settling time. Consequently, this paper considers problems where a simplistic DMC or GPC implementation may not be so effective; for instance problems with:

(1) Poor open-loop dynamics.
(2) State or output constraints.

In these cases, DMC or GPC with an input horizon of one may produce closed-loop behaviour close to the open-loop and therefore unsatisfactory. State constraints may also severely restrict the operating region and have a strong influence on the constrained control law.

A common objective is to guarantee asymptotic stability and recursive constraint satisfaction for a set of initial states that is as large as possible and with both a minimal control cost and computational load and to minimize control performance. In this paper it will be assumed that asymptotic stability is taken for granted Mayne et *al* (2000) if one uses a dual-mode prediction with an appropriate performance index including the terminal weight. Consequently, the main outstanding issues are:

(1) maximising the feasible region, that is the region in the state space for which the control law is defined and meets the terminal constraints [1].
(2) maintains a sensible limit on the implied online computational load.
(3) obtaining good enough closed-loop performance.

One simplistic approach Tan et *al* (1992) simply defines a large number of alternative linear control laws offline and then selects online from the currently feasible laws (i.e. the current state lies with the associated MAS), the one giving best performance. This approach is also easily extended to the robust case Kothare et *al* (2003). However, a major weakness is that the optimum constrained control law is known to be linear time varying Bemporad et *al* (1996); Rossiter et *al* (2005) and thus this approach can given suboptimal performance when feasible and may also give significant restrictions to feasibility. Moreover, although this is perhaps less of an issue for some processes, the MAS definition, especially in the robust case, can require a large number of linear inequalities, and thus storage and set-membership tests may become non-trivial for more than a few alternative control laws. The same issue is well understood within parametric programming.

Other work has looked at alternative ways of formulating the degrees of freedom for optimisation, for instance by interpolation methods Bacic et *al* (2003); Rossiter et *al* (2004). This paper will not pursue that angle as current proposals do not extend well to large dimensional systems and also by sticking with a conventional DMC or GPC paradigm, the results have greater potential for take up by colleagues and industry.

Hence, the main proposal in this paper is suprisingly simple. Use a conventional dual-mode algorithm such as in Rossiter et *al* (1998); Scokaert et *al* (1998), but critically with one major difference; do not assume a match between the underlying terminal control law and what would arise from unconstrained minimisation of the performance index. In fact, this idea is implicit in DMC with large horizons which is equivalent to a dual mode algorithm where the terminal law has a gain of zero and thus clearly does not match the unconstrained control law. Thus a main contribution of this paper is insight and to demonstrate the potential in an algorithm that recently has been somewhat neglected. Specifically it will be shown

[1] Hereafter this region is assumed to be the maximal admissible set (MAS) Kolmanovsky and Gilbert et *al* (1996); its shape and volume depends upon both constraints and the closed-loop dynamics.

that using a detuned terminal control law in conjunction with a performance index that should give a more highly tuned control law in fact loses very little by performance, even for minimal control horizons and can gain hugely by way of feasible regions.

A further minor issue is how to present the results. Many papers restrict themselves to two-state and perhaps three-state systems because feasible regions can be plotted clearly. For higher dimensions, projections can be used but the number required grows rapidly with dimension and give a restricted view anyway. Here we propose an alternative and very simple approach that can be used for any dimension of system if suitably combined with Monte Carlo type approaches and gives very easy to interpret figures.

Section 2 will give some background to MPC, the modelling assumptions and then gives a detailed summary of conventional predictive control algorithm. Section 3 introduces the proposed algorithm and associated information and Section 4 gives some simple numerical examples. Section 5 gives higher order examples and illustrates the alternative method of comparing different control schemes for feasibility and performance. The paper finishes with conclusions and plans for future work.

2. BACKGROUND

This section introduces standard material from the existing literature on MPC and invariant sets and a conventional dual mode optimal control OMPC is defined.

2.1 Model and performance objective

This paper considers linear systems of the form

$$x(k+1) = Ax(k) + Bu(k), \qquad k = 0, \ldots, \infty \tag{1}$$

and subject to constraints

$$u(k) \in \mathcal{U} \equiv \{u : \underline{u} \leq u \leq \overline{u}\}, \qquad k = 0, \ldots, \infty, \tag{2a}$$

$$x(k) \in \mathcal{X} \equiv \{x : \underline{x} \leq x \leq \overline{x}\}, \qquad k = 0, \ldots, \infty. \tag{2b}$$

$x(k) \in \mathbb{R}^{n_x}$ and $u(k) \in \mathbb{R}^{n_u}$ denote state and input vectors at discrete time k with n_x and n_u respectively denoting the number of states and inputs of the system. More general linear state, input and mixed state/input constraints can also be considered without significantly complicating further sections.

A typical performance index is based on a 2-norm and is computed over infinite horizons for both the input and output predictions [2]. The cost is equivalent to one used in optimal control with the main tuning parameters being the matrix weights Q, R:

$$J = \sum_{k=0}^{\infty} (x(k)^{\mathrm{T}} Qx(k) + u(k)^{\mathrm{T}} Ru(k)) \tag{3}$$

with $Q \in \mathbb{R}^{n_x \times n_x}$ and $R \in \mathbb{R}^{n_u \times n_u}$ positive definite state and input cost weighting matrices.

The control law is defined as that which minimises the predicted value of J using the allowed flexibility in the future control moves $u(k), k = 0, 1, ...$ and subject to constraints (2).

[2] In DMC and GPC the input stops changing after n_u steps so this is equivalent to an infinite horizon anyway.

Remark 2.1. As this paper is focusing on concepts, this cost does not include integral action and tracking, however inclusion is straightforward, e.g. Rossiter (2006) at the cost of an increased state dimension.

2.2 OMPC

OMPC is the algorithm of Rossiter et *al* (1998); Scokaert et *al* (1998) and is often taken as a standard benchmark in the literature. The key idea here is to embed into the predictions the unconstrained optimal behaviour and optimise about this. Consequently, the algorithm will find the global optimal, with respect to (3), whenever that is feasible.

To allow for the case where the unconstrained optimal predictions are infeasible, that is violate constraints (2), some control perturbations $c(k)$ are allowed over a control horizon n_c; these constitute the degrees of freedom (d.o.f.) within the optimisation. Hence the input predictions are defined as follows:

$$\begin{aligned} u(k+i) &= -Kx(k+i) + c(k+i); \quad i = 0, ..., n_c - 1 \\ u(k+n_c+i) &= -Kx(k+n_c+i); \quad i \geq 0 \end{aligned} \tag{4}$$

with K the optimum state feedback which minimises J of (3). The d.o.f. for optimisation are summarised in vector $C = [c(k)^T, ..., c(k+n_c-1)^T]^T$.

Given that feedback K is defined as the optimal control corresponding to J, it is easy to show Rossiter (2003) that optimisation of J over input predictions (4) is equivalent to minimising $J = C^T W C$ for a suitable positive definite W ($W = B^T \Sigma B + R$, $\Sigma - \Phi^T \Sigma \Phi = Q + K^T RK$, $\Phi = A - BK$) and thus, in the absence of constraints, the optimum is $C = 0$. Where the unconstrained predictions would violate constraints, non-zero C would be required to ensure constraints are satisfied.

2.3 Constraint handling, MAS and MCAS

It is known that for suitable M, N, d (e.g. Rossiter (2003)), the input predictions (4) and associated state predictions for model (1) satisfy constraints (2) if:

$$Mx + NC \leq d \tag{5}$$

Remark 2.2. The MCAS (maximal controlled admissible set) is defined as $S = \{x : \exists C \text{ s.t. } Mx + NC \leq d\}$. The volume and shape of S depends on M, N, d which vary with the state-feedback K within (4) and the model (1). Critically we note here that S does not depend in any way on J.

Remark 2.3. It is assumed throughout this paper that any sets used are not only invariant Blanchini (1999), but in general are the maximal admissible sets (MAS, Gilbert et *al* (1991)) corresponding to any given prediction class. It is known therefore that using such sets gives a guarantee of recursive feasibility, which in combination with a cost based on infinite horizons is sufficient to establish a guarantee of convergence. There are some nuances for the uncertain case (e.g. Pluymers et *al* (2005)) but these are not central to this paper.

Algorithm 2.1. The OMPC algorithm is summarised as:

$$C = \arg \min_{C} C^T W C \quad \text{s.t.} \quad Mx + NC \leq d \tag{6}$$

Use the first element of C in the control law of (4).

2.4 Summary

There are two key observations we wish to emphasis at this point.

(1) The feasible region S depends only on the prediction class chosen. For conventional MPC algorithms this class is defined by the choice of n_c (number of free moves) and by K, the terminal control.
(2) The performance measure J is actually distinct from the prediction class. A well conditionned optimisation problem and the desire for a global optimum would suggest a synergy between J and K Scokaert et *al* (1998).

This paper seeks to explore more carefully the potential gains of not having a snyergy between the prediction class and the performance index. Although it is intuitively obvious that the optimisation is not as well-posed, this paper will demonstrate through examples that one can achieve significant feasibility gains (increases in the volume of the MCAS) with very small loses in performance.

3. A MPC ALGORITHM EXPLOITING NON-SYNERGY

This section develops some of the algebra required for an dual-mode MPC control law that combines a prediction class with good feasibility, but hence potentially detuned performance, with a performance index that indicates a desire for high performance.

Definition 3.1. (Performance). Assume that the optimal performance and hence both predictions and simulations are all assessed by the same cost function:

$$J = \sum_{k=0}^{\infty} x(k+1)^T Q x(k+1) + u(k)^T R u(k) \qquad (7)$$

Definition 3.2. (Prediction classes). Assume that there are two alternative prediction classes[3]. Let these prediction classes be given as:

OMPC

$$\begin{array}{ll} u(k+i) = -K_1 x(k+i) + c(k+i); & i = 0, ..., n_c - 1 \\ u(k+n_c+i) = -K_1 x(k+n_c+i); & i \geq 0 \end{array}$$

OMPCB

$$\begin{array}{ll} u(k+i) = -K_2 x(k+i) + c(k+i); & i = 0, ..., n_c - 1 \\ u(k+n_c+i) = -K_2 x(k+n_c+i); & i \geq 0 \end{array} \qquad (8)$$

where K_1 is the feedback minimising J in the unconstrained case and K_2 is a detuned feedback giving a large feasible region. For the purposes of this paper, K_2 is selected by finding the optimal state feedback for a cost of the same form as J but with a significantly larger input weighting R. However, other methods of defining K_2 to give good feasibility are also viable.

3.1 Summary of OMPCB controller

The OMPCB algorithm has a more complicated performance index than OMPC. Following substitution of the prediction class (8) and system model (1) into J (7) it is apparent that this index takes the form:

$$J_2 = C^T W_2 C + C^T W_3 x + x^T W_4 x \qquad (9)$$

for suitable W_2, W_3 (see chaps 6,7 of Rossiter (2003)). The key point for the reader to note is that unlike for OMPC, $W_3 \neq 0$ and in fact W_2 is also more complex in structure than W in (6). The last term is ignored as not being dependent on the d.o.f. C.

The constraint inequalities which ensure that the OMPCB predictions meet system constraints are given as:

$$M_2 x + N_2 C \leq d_2 \qquad (10)$$

Algorithm 3.1. The OMPCB algorithm is summarised as:

$$C = \arg \min_C C^T W_2 C + C^T W_3 x \quad \text{s.t.} \quad M_2 x + N_2 C \leq d_2 \qquad (11)$$

Use the first element of C in the OMPCB control law of (8) to compute $u(k)$.

4. USING CONTOUR PLOTS TO ILLUSTRATE THE EFFICACY OF OMPCB

This section will give a simple illustration, using two-state examples, of how to decide whether OMPC or OMPCB is to be preferred, solely on the basis of closed-loop performance. For now, we consider only points inside the feasible region of all algorithms compared although it will be clear that OMPCB, by design, has a far better feasibility than OMPC and thus would automatically win any feasibility comparison.

The initial idea (which is made more efficient later) is to grid the entire feasible state space and for each point compute the run-time cost for closed-loop simulations beginning at that point. The run time cost is given by computing the cost J for the actual closed-loop state and input evolutions. The resultant function $J(x)$ is known to be piecewise quadratic and convex. Viewing of $J(x)$ for the two state case is made easier by drawing contours, on the state-space, of initial points x which give the same run-time cost.

Two different strategies are compared: (i) OMPC and (ii) OMPCB. For both algorithms we use $n_c = 2$ as it is common industrial practise to minimise over just a few control moves. Moreover, if n_c is large then the two algorithms will become equivalent, to within engineering precision, anyway.

4.1 A two-state example

The model and constraints are given by :

$$A = \begin{bmatrix} 1 & 0.1 \\ 0 & 1 \end{bmatrix}, \qquad B = \begin{bmatrix} 0 & 0.0787 \\ 0.0787 & 0 \end{bmatrix} \qquad (12)$$

$$C = \begin{bmatrix} 1.7993 & 13.2160 \\ 0.8233 & 0 \end{bmatrix}, \quad D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \qquad (13)$$

$$\overline{u} = [1,2]^{\mathrm{T}}, \qquad \underline{u} = -[1,2]^{\mathrm{T}} \qquad (14)$$

$$\overline{x} = [100,100]^{\mathrm{T}}, \qquad \underline{x} = [-100,-100]^{\mathrm{T}} \qquad (15)$$

The LQR-optimal controller is derived with $Q = \mathrm{diag}(1,0)$, $R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. The controller 1 is $K_1 = \begin{bmatrix} 1.2896 & 11.6776 \\ 2.3865 & 1.5282 \end{bmatrix}$, and the controller 2 is $K_2 = \begin{bmatrix} 1.0236 & 8.0573 \\ 0.7918 & 1.1028 \end{bmatrix}$ has R=10.

[3] Naturally one could have more, but two is enough to demonstrate the concepts.

4.2 Run-time costs and interpretation

The contours for run-time costs of 1,10,20,50 are given in figure 1 overlaid on the MCAS for OMPC and OMPCB.

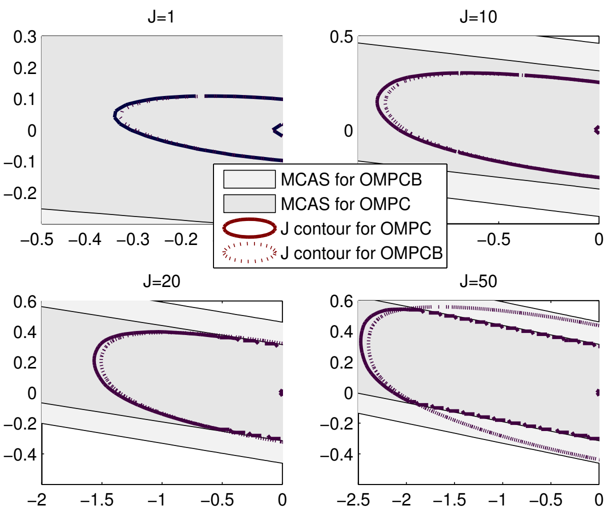

Fig. 1. Contour plot comparison of OMPC and OMPCB different controllers at levels $J = 1, 10, 20, 50$

Two observations are clear:

- For states well within the MCAS, OMPC gives slightly better performance than OMPCB. This is clear as the contour for the same value of $J(x)$ is further out from the origin.
- As the initial state gets closer to the boundary of the MCAS for OMPC, the OMPCB algorithm has better performance, that is its contour for the same $J(x)$ is now further out.

Critically however, the performance loss for OMPCB is relatively small whereas the feasibility gain is huge.

Remark 4.1. In principle the same algorithm as proposed in Tan et *al* (1992) could be deployed: if the state is close to the origin, use OMPC and if not use OMPCB. However, given the complexity of $J(x)$ one would be unlikely to determine this explicity as in parametric solutions and thus the potential and detailed implementation of such an algorithm is left for future consideration.

5. EXTENSIONS OF COMPARISONS TO HIGHER DIMENSIONS

Although the contour plots work well enough for systems with two-states, there is no simple way of extending this to large systems. Yet, of particular interest to a potential user is a clear comparison of how OMPC and OMPCB compare over the entire MCAS for higher dimensional systems. This section proposes a novel yet very simple way of displaying the relevant feasibility and performance information that does extend to arbitrary dimensions, does not require exhaustive computation over the entire phase space, yet gives a very insightful summary. First we illustrate the basic principle using the two state example of the previous section and then apply this to three and four state examples.

5.1 The two-state case

This section gives a proposal for how to compare feasibility and performance of different algorithms; although in essence trivial, we believe this approach is novel within the literature and potentially of great use. The procedure is summarised as follows:

(1) Take the MCAS for OMPC, OMPCB (e.g. as given in figure 1) and choose an arbitrary search direction, for instance in figure 2, direction 1 is $x = [1 \quad 0]^T$.
(2) Scale this direction until it intersects with the boundary of the MCAS for OMPCB; hence define the boundary point as $w = \mu x$. This is illustrated in figure 2 which shows $w \approx [3.5\ 0]^T$ for direction 1.
(3) Compute $J_1(\lambda w), J_2(\lambda w), \quad \forall\lambda,\ 0 \leq \lambda \leq 1$ which captures all feasible states for OMPCB along the chosen direction. If OMPC is infeasible for some λ, set $J_1(\lambda w) = 0$. Plot J_1, J_2 against λ [see figure 3a]. This figure clearly shows for what ranges of λ and hence initial states x in the given direction, either OMPC or OMPCB gives better performance and feasibility. [In this case there is little difference in performance when both are feasible but OMPC has far better feasibility.]
(4) Plot J_2/J_1 against λ where in this instance infeasibility is marked with $J = \infty$ [figure 3b]. In this case values greater than one indicate the ranges of λ such that OMPC is best, values less than 1 for which OMPCB is best, values of zero indicate that OMPC is infeasible but OMPCB is feasible. [In this case we observe that OMPCB has near equivalent performance to OMPC for small λ and better where OMPC is near to its feasibility boundary.]

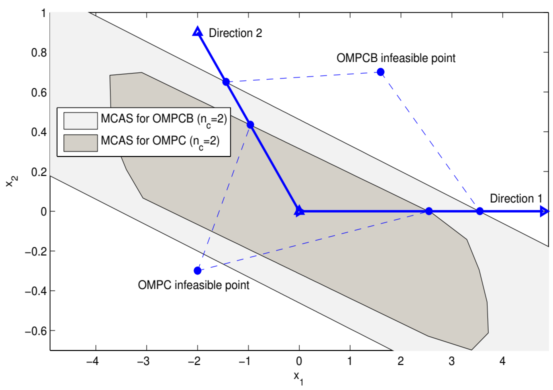

Fig. 2. Search directions, MCAS and w for two state example

Figure 2 gives an illustration for a single direction in the state space. In reality a comparison is needed for all possible directions, for example figure 2 demonstrates an alternative, direction 2. Ideally one should compute figures analogous to figure 3 for many different directions which span the space. In this paper for instance, in the interest of economy, twenty evenly spread directions will be used hereafter; a more formal generalisation could use some form of stochastic approach to capture sufficient directions.

- The plots of $(J_1,\ J_2)$ *vs* λ for several different directions are overlaid on the same plot (e.g. figure

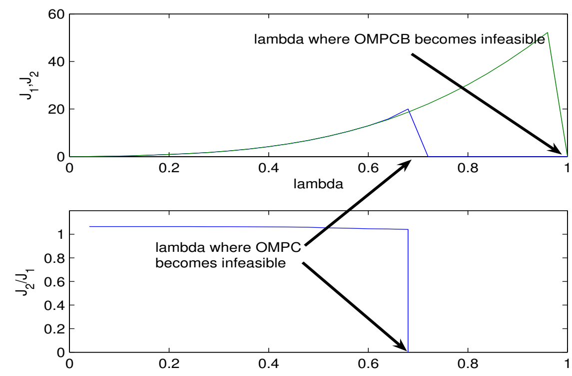

Fig. 3. Performance and feasibility comparisons over a single search direction for two-state system

4a) as this gives an excellent overview of both the cost and feasibility comparison across the entire space. Note that due to the large variance in values of J for different directions, it was found beneficial to use a log scale.

- The plots of J_2/J_1 are also overlaid (e.g. figure 4b) as this gives an alternative but useful insight into the same comparison. Notably:
  - if a line drops to zero for small λ, then the feasibility benefit of OMPCB is huge.
  - the y-axis value of one is a clear marker of whether OMPC (greater than 1) or OMPCB (less than 1) is performing better.

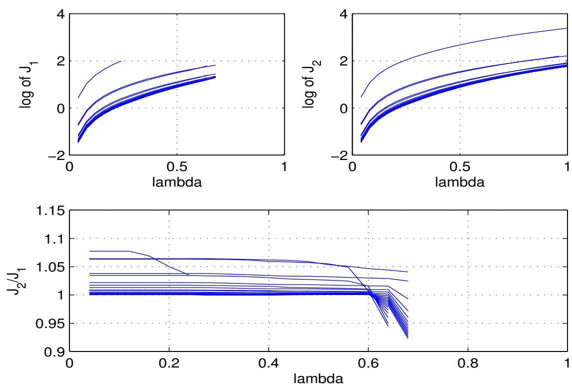

Fig. 4. Performance and feasibility comparisons over 20 search directions for two-state system

5.2 Three and four state examples

Figures 4, 5, 6 show the plots $log(J_1, J_2)$ vs λ and J_2/J_1 vs λ for 20 alternative directions for three different examples. It is clear that these plots give a quick and easy view of which strategy is to be preferred. In fact one could argue that for each of these examples, OMPCB has similar or better performance when OMPC is feasible in addition to a markedly larger feasible region. In some directions OMPCB has feasibility as much as five times further from the origin while giving almost identical performance close in.

Example 3 This model and constraints are given by:

$$A_3 = \begin{bmatrix} 0.9146 & 0 & 0.0405 \\ 0.1665 & 0.1353 & 0.0058 \\ 0 & 0 & 0.1353 \end{bmatrix}, B_3 = \begin{bmatrix} 0.0544 & -0.0757 \\ 0.0053 & 0.1477 \\ 0.8647 & 0 \end{bmatrix} \tag{16}$$

$$C_3 = \begin{bmatrix} 1.7993 & 13.2160 & 0 \\ 0.8233 & 0 & 0 \end{bmatrix}, D_3 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{17}$$

$$\overline{u} = [1, 2]^{\mathrm{T}}, \qquad \underline{u} = -[1, 2]^{\mathrm{T}} \tag{18}$$

$$\overline{x} = [100, 100, 100]^{\mathrm{T}}, \quad \underline{x} = [-100, -100, -100]^{\mathrm{T}} \tag{19}$$

The LQR-optimal controller is derived with $Q = \mathrm{diag}(1, 0)$ and $R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. The feasible controller has $R = 10I$.

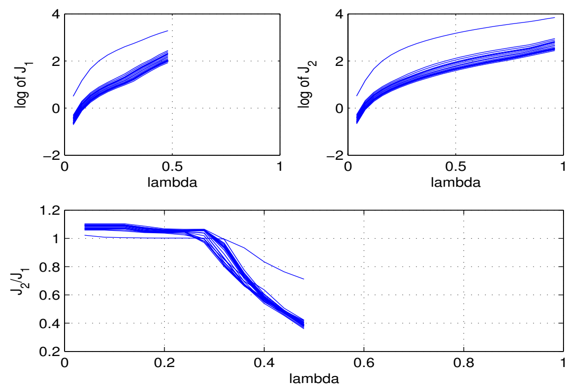

Fig. 5. Performance and feasibility comparisons over 20 search directions for three-state system

Example 4 This model and constraints are given by :

$$A_4 = \begin{bmatrix} 0.9146 & 0 & 0.0405 & 0.1 \\ 0.1665 & 0.1353 & 0.0058 & -0.2 \\ 0 & 0 & 0.1353 & 0.5 \\ 0 & 0 & 0.1353 & 0.8 \end{bmatrix}, \tag{20}$$

$$B_4 = \begin{bmatrix} 0.0544 & -0.0757 \\ 0.0053 & 0.1477 \\ 0.8647 & 0 \\ 0.5 & 0.2 \end{bmatrix} \tag{21}$$

$$C_4 = \begin{bmatrix} 1.7993 & 13.2160 & 0 & 0.1 \\ 0.8233 & 0 & 0 & -0.3 \end{bmatrix}, D_4 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{22}$$

$$\overline{u} = [1, 2]^{\mathrm{T}}, \qquad \underline{u} = -[1, 2]^{\mathrm{T}} \tag{23}$$

$$\overline{x} = [100, 100, 100, 100]^{\mathrm{T}}, \quad \underline{x} = [-100, -100, -100, -100]^{\mathrm{T}} \tag{24}$$

The LQR-optimal controller is derived with $Q = \mathrm{diag}(1, 0)$ and $R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ abd the OMPCB has $R = 10I$.

6. CONCLUSION AND FUTURE WORK

This paper has given careful consideration and novel insights into the tradeoffs between feasibility and performance within dual mode MPC. Although it is unsurprising that one can always achieve better performance near the origin with a well tuned terminal control law, that is one that has snyergy with the performance index, this

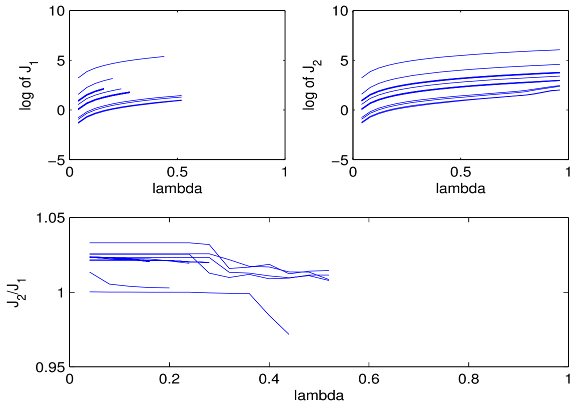

Fig. 6. Performance and feasibility comparisons over 20 search directions for four-state system

paper has sought to give a better understanding of the extent of the feasible region for which this might hold true. Specifically it is shown that, especially as one nears the feasibility boundary, that is constraints become active, the choice of an optimal terminal feedback my be very little better and indeed may even be worse than an alternative more detuned choice.

The authors do not believe that a generic result exists, however a novel means of presenting the feasibility versus performance trade off has been proposed that can be computed very quickly for any given example, notably including examples of large state dimension, and thus allow the designer to take informed decisions.

It is noted that the proposed graphic may serve as a useful start point for combining the work of this paper with Tan et *al* (1992) and this is one possible avenue of future work. Also, although this paper has focussed on the LTI case, extensions to the robust case are necessary, but are expected to be straightforward following the work of Pluymers et *al* (2005). Finally, we should note that a formal comparison with the recently proposed alternative of triple mode MPC Imsland et *al* (2005) would be worthwhile.

7. ACKNOWLEDGMENTS

This principal investigator is funded by ORS, the University of Sheffield and the ACSE department.

REFERENCES

Bacic, M. and Cannon, M. and Lee, Y. I. and Kouvaritakis, B., General Interpolation in MPC and Its Advantages, IEEE Transactions on Automatic Control, 48, 6, June, 1092–1096, 2003

Bemporad, A., M. Morari, V. Dua and E.N. Pistokopoulos, The explicit linear quadractic regulator for constrained systems, Automatica, 38(1), 3-20, 2002.

Blanchini, F., Set Invariance in Control, Automatica, 1999, 35, 1747-1767

Camacho, E. and C. Bordons, Model predictive Control, Springer Verlag, 2005

Clarke, D.W., C. Mohtadi and P.S. Tuffs, Generalised predictive control, Parts 1 and 2, Automatica, 23, 137-160, 1987.

Cutler C.R. and B.L. Ramaker, Dynamic matrix control - a computer control algorithm, Proceedings American Control Conference, 1980.

Ding, Y., and Rossiter, J.A., Enlarging Feasible Regions for Interpolation Methods, UKACC, 2006

Gilbert, E. G. and Tan. K. T., Linear Systems with State and Control Constraints : The Theory and Application of Maximal Output Admissable Sets, IEEE Transactions on Automatic Control, 36, 9, 1008-1020, 1991

Imsland, L. and J.A. Rossiter, Time varying terminal control, IFAC world congress, 2005

Kothare, M. V. and Balakrishnan, V. and Morari, M., Robust Constrained Model Predictive Control using Linear Matrix Inequalities, Automatica, 32, 1361-1379, 1996

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

Mendez, J.A., B. Kouvaritakis and J.A. Rossiter, State Space approach to interpolation in MPC, International journal of robust nonlinear control, 10, 27-38, 2000

Pluymers, B. and Rossiter, J.A. and Suykens, J.A.K. and De Moor, B., , IFAC world congress, 2005

Richalet, J., Rault, A., Testud, J.L. and J. Papon, Model predictive heuristic control: applications to industrial processes, Automatica, 14(5), 413-428, 1978.

Rossiter, J.A., M.J.Rice and B. Kouvaritakis, A numerically robust state-space approach to stable predictive control strategies, Automatica, 34, 1, pp65–73, 1998

Rossiter, J.A. and Kouvaritakis, B. and Bacic, M., Interpolation based computationally efficient predictive control, International Journal of Control, 77, 290-301, 3, 2004

Rossiter, J.A., B. Kouvaritakis, M. Cannon, An algorithm for reducing complexity in parametric predictive control, IJC, 78, 18, pp1511-1520, 2005

Rossiter, J.A., Model based predictive control: a practical approach, CRC Press, 2003

Rossiter, J.A., A global approach to feasibility in linear MPC, ICC, 2006

Rossiter, J.A. and B. Pluymers, The potential of interpolation for simplifying predictive control and application to LPV systems, Workshop on NMPC, Stuttgart, 2005

Scokaert, P.O.M. and Rawlings, J.B., Constrained linear quadratic regulation, IEEE Transactions on Automatic Control, 43, 8, 1163-1168, 1998

Tan, K.T. and E.G. Gilbert, Multimode controllers for linear discrete time systems with general state and control constraints, in Optimisation Techniques and Applications (World Scientific, Singapore), pp433–442, 1992.

Kolmanovsky and Gilbert, Multimode Regulators for Systems with State and Control constraints and disturbance Inputs, 2006

Wan, Z. and Kothare, M. V., Robust output feedback model predictive control using off-line linear matrix inequalities, Journal of Process Control, 12, 763-774, 2002