Reconfiguration and Scheduling in Flight Using Quasi-LPV High-Fidelity Models and $MBPC$ Control

Mihai Huzmezan * Jan Maciejowski
Department of Engineering, University of Cambridge,
Cambridge CB2 1PZ, United Kingdom.

Abstract

This paper describes advances towards reconfiguration and scheduling in flight control systems using high fidelity models expressed in a quasi-LPV form, Fault Detection and Isolation (FDI), model approximation/simplification and constrained Model Based Predictive Control ($MBPC$). The strategy is applied to a well known missile example.

Keywords: Flight Control, Reconfigurable Control, Model Based Predictive Control ($MBPC$).

1. Introduction

Several apparently unconnected research topics are mentioned in the title. The main contribution of this paper is their integration in a framework which provides a solution for scheduling and reconfiguration using Model Based Predictive Control ($MBPC$), relying on quasi-LPV internal models.

We approach reconfigurable control as a multi-model adaptive control problem, namely one in which adaptation occurs by discrete changes to the control algorithm, rather than continuous tracking of a gradually-changing model. Scheduling as part of the adaptation has already been proved successful in flight control applications [5, 2].

There has been some progress in analysing the stability of certain multi-model control schemes [8, 7], although it is too early to know whether these approaches will be helpful for our proposal.

The $MBPC$ control strategy can be formulated in multi-model setting. This feature together with its special ability to handle constraints [11, 4] and its inherent self-reconfiguration property [6] made us to consider it as a potential control tool for reconfigurable control.

Patcher et. al. [9] were the first to view a simplified version of constrained predictive control as a promising tool for reconfiguration. Their scheme was an indirect adaptive approach with the system identification module employing a mixed identification method (optimisation involving flight mechanics information) and the controller being split into two loops (an LQR inner loop and an outer linear programme on-line optimisation to enforce constraints a step ahead).

We propose to represent the controlled plant by a quasi-LPV model which represents a high-fidelity nonlinear model. We therefore do not use an identification module, at least for ordinary 'scheduling'. For reconfiguration in case of failures some fault detection module is required; for the purpose of this work we assume the existence of such a module.

We consider that the particular combination of high fidelity models, fault detection and isolation (FDI), model approximation/simplification and constrained predictive control enables a generic solution to reconfiguration and adaptation [6].

The paper will firstly introduce quasi-LPV models and summarise the $MBPC$ technique. Next it shows how these can be integrated to give a strategy for scheduling and reconfiguration. The solution will be illustrated via a well known missile example.

2. High fidelity models written in the quasi-LPV form

High fidelity dynamic models are increasingly built for complex plants. This has been the case in the aerospace industry for many years.

In [12] a quasi-LPV model that embeds all the plant nonlinearities without interpolating between point-wise (Jacobian) linearisations is presented. The main characteristic of these models compared with the usual LPV way of representing systems is that the scheduling variable is a state of the model.

Such a way of modelling the plant gives the clarity required by industry and eases certification since it retains a physical meaning for the elements of the model LTV matrices. It is advisable to have the scheduling variable as a system output rather than an estimate.

The quasi-LPV approach is mostly suited for systems exhibiting output nonlinearities (*e.g.* aerospace applications). When it is impossible to embed all the system nonlinearities in the output then the transformations used in producing the quasi-LPV model, see equation (3), have to be approximated up to a first order terms in all the states except the scheduling parameters.

To develop the quasi-LPV model we start with a nonlinear missile model with the states α (angle of attack) and q (pitch rate), and the command δ (horizontal fin angle) of which complete details are given in [12].

*e-mail: mh2@eng.cam.ac.uk

Then we cast the nonlinear model such that the nonlinearities depend only on the scheduling variable α:

$$\frac{d}{dt}\begin{bmatrix}\alpha\\ q\end{bmatrix} = f(\alpha) + \begin{bmatrix}A_{11}(\alpha) & A_{12}(\alpha)\\ A_{21}(\alpha) & A_{22}(\alpha)\end{bmatrix}\begin{bmatrix}\alpha\\ q\end{bmatrix} + \begin{bmatrix}B_{11}(\alpha)\\ B_{21}(\alpha)\end{bmatrix}\delta \quad (1)$$

A family of equilibrium states, parametrised by the scheduling variable α, is obtained by setting the state derivatives to zero:

$$0 = f(\alpha) + A(\alpha)\begin{bmatrix}\alpha\\ q_{eq}(\alpha)\end{bmatrix} + B(\alpha)\delta_{eq}(\alpha)$$

Providing that there exist continuously differentiable functions $q_{eq}(\alpha)$ and $\delta_{eq}(\alpha)$, we are able to write the system (1) in the following form [12]:

$$\begin{aligned}&\tfrac{d}{dt}\begin{bmatrix}\alpha\\ q - q_{eq}(\alpha)\end{bmatrix} = \\ &\begin{bmatrix}0 & A_{12}(\alpha)\\ 0 & A_{22} - \frac{d}{d\alpha}q_{eq}(\alpha)A_{12}(\alpha)\end{bmatrix}\begin{bmatrix}\alpha\\ q - q_{eq}(\alpha)\end{bmatrix} + \\ &\begin{bmatrix}B_{11}(\alpha)\\ B_{21}(\alpha) - \frac{d}{d\alpha}q_{eq}(\alpha)B_{11}(\alpha)\end{bmatrix}(\delta - \delta_{eq}(\alpha))\end{aligned} \quad (2)$$

This form gives a different α-dependent family than would be obtained by pointwise linearisation, however, the state matrix A at a given parameter value will still retain the modes.

In order to use (2), the function $\delta_{eq}(\alpha)$ must be known. This can be estimated by using an 'inner loop' [12] but, because of model uncertainty, this can reduce the robustness of the main control loop in a way which is difficult to predict at the design stage. Like in [12] we avoid the problem generated by the existence of an inner loop required to compute $\delta_{eq}(\alpha)$ by adding an integrator at the plant input. As a result we have the following form for the system dynamics:

$$\begin{aligned}&\tfrac{d}{dt}\begin{bmatrix}\alpha\\ q - q_{eq}(\alpha)\\ \delta - \delta_{eq}(\alpha)\end{bmatrix} = \\ &\begin{bmatrix}0 & A_{12}(\alpha) & B_{11}(\alpha)\\ 0 & A_{22} - \frac{d}{d\alpha}q_{eq}(\alpha)A_{12}(\alpha) & B_{21}(\alpha) - \frac{d}{d\alpha}[q_{eq}(\alpha)]B_{11}(\alpha)\\ 0 & -\frac{d}{d\alpha}[\delta_{eq}(\alpha)]A_{12}(\alpha) & -\frac{d}{d\alpha}\delta_{eq}(\alpha)B_{11}(\alpha)\end{bmatrix} \times \\ &\begin{bmatrix}\alpha\\ q - q_{eq}(\alpha)\\ \delta - \delta_{eq}(\alpha)\end{bmatrix} + \begin{bmatrix}0\\ 0\\ 1\end{bmatrix}\nu\end{aligned} \quad (3)$$

This final form is actually the representation used for control purposes. If α remains constant, the $MBPC$ controller stabilising this model will drive the states to zero which means that the δ input to the plant will be set at the true δ_{eq} trim value. The $MBPC$ internal model will be the corresponding LTI system obtained at each value of the scheduling parameter α.

The controlled output of the plant is the normal acceleration (n_Z), which is a nonlinear function of α and $\delta - \delta_{eq}(\alpha)$. In the internal model a linearised approximation of n_Z with respect to α is used, as in [12].

The normal force (C_Z) and pitch moment (C_m) aerodynamic coefficients are embedded in the expressions for $q_{eq}(\alpha)$, $\delta_{eq}(\alpha)$, $A(\alpha)$ and $B(\alpha)$. The representation of C_Z and C_m is approximated by the following expressions:

$$\begin{aligned}C_Z &= \Phi_Z(\alpha) + b_Z\delta\\ C_m &= \Phi_m(\alpha) + b_m\delta\end{aligned}$$

where the expressions for b_Z and b_m are given in [12], and $\Phi_Z(\alpha)$, $\Phi_m(\alpha)$ are polynomials in α. Let k_Z and k_m, respectively, be the vectors of coefficients of these polynomials. We assume that k_Z and k_m will be the outputs of an FDI module, as shown in Figure 1. These expressions for C_Z and C_m explicitly contain contributions from all the missile's surfaces, and are based on data obtained from wind tunnel tests. Such data may be available in the form of a database [3].

3. The $MBPC$ multiple model control strategy

The defining feature of $MBPC$ is the repeated optimisation of a performance objective over a finite horizon. Given a set-point, a reference $r(k+l)$ is produced and used within the optimisation of the cost function (4). Manipulating the control variable $u(k+l)$, over the control horizon (N_u), the algorithm drives the predicted output $\tilde{y}(k+l)$, over the prediction horizon (N_y), towards the reference.

$$J(k) = \sum_{l=1}^{N_y}\|(\tilde{y}(k+l) - r(k+l)\|^2_{Q(l)} + \sum_{l=0}^{N_u}\|\Delta u(k+l)\|^2_{R(l)} \quad (4)$$

The minimisation is performed subject to constraints on:

- the inputs levels: $u_l(l) \le u(l) \le u_u(l)$ where $k \le l \le k + N_u - 1$
- the input rates of change: $\Delta u_l(l) \le \Delta u(l) \le \Delta u_u(l)$ where $k \le l \le k + N_u - 1$
- the output (and state) levels: $y_l(l) \le \tilde{y}(l) \le y_u(l)$ where $k \le l \le k + N_y$

In normal operation the weights $Q(l)$ and $R(l)$ are independent of k, but if a failure occurs they may need to vary with k. The norm $\|.\|^2_Q$ within the cost function is defined as $\|x\|^2_Q = x^TQx$. For prediction it is assumed that $\Delta u(l) = 0$ for $l \ge k + N_u$. The prediction model is based on the linear frozen representation of the quasi-LPV model. As formulated, the optimisation is a quadratic programming (QP) problem, and can be solved using standard algorithms.

The predictor employing the internal model, based on the measurement of the current state, provides the optimiser with the future predicted values of states and outputs. The optimiser contains the cost function (4) involving adjustable weights if necessary with the main task of computing the present control move such that the predicted output follows the reference in the desired manner.

4. A new reconfiguration strategy

The four system components shown in Figure 1 and referred to in Section 1., work together as follows:

1. When a failure occurs the FDI system pinpoints the nature of the failure. Updated k_Z and k_m parameters are passed to the quasi-LPV high fidelity model. Actuator failure information is passed to the $MBPC$ controller as constraint modifications, if appropriate.

2. Freezing the quasi-LPV model with respect to the parameter vector yields a linear model which is used by the $MBPC$ as its internal model. An important advantage for real-time implementation is that this is a computationally inexpensive way of obtaining an adaptive internal model, compared with other possibilities such as identification.

3. The constrained $MBPC$ controller provides inputs to the plant. Giving it enough degrees of freedom (a large enough set of control inputs) enables it to keep the plant close to the required trajectory. We are assuming here that any failures are compatible with maintaining this trajectory. Changing the required trajectory is a higher-level problem which we do not address in this paper.

All the components performing the functions mentioned above are shown in Figure 1. We claim that the system ensures the following successive layers of reconfiguration:

- A simple reconfiguration occurs if there are actuator failures which are consistent with maintenance of the required flight conditions. This can occur even without FDI information [6].

- Small structural failures are reflected in the high-fidelity model, as shown in Section 2., and hence in the linear internal model used by the $MBPC$ controller. No tuning is necessary for the controller; its original cost function still reflects well the various criteria for which it was designed.

- Major failures will again lead to a change in the linear internal model, but now re-tuning of the controller will be necessary. We are currently investigating this; an approach which keeps the horizons fixed but adjusts the weights in (4) appears to be the most promising at present.

5. The autopilot structure

In Figure 1 all the essential components of the system which perform the functions referred in the Section 4. are depicted.

The quasi-LPV model runs in parallel with the nonlinear plant providing the states equilibrium values $q_{eq}(\alpha)$, $\delta_{eq}(\alpha)$ together with the $A(\alpha)$, $B(\alpha)$, C, D matrices depending on the current value of the parameter α.

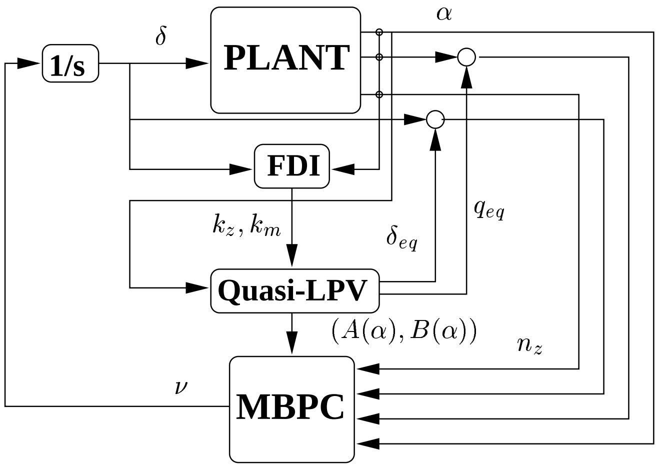

Figure 1: The reconfiguration and scheduling $MBPC$ based strategy

The structure of the $MBPC$ controller consists of standard modules. States, including α, are assumed measurable, since they are usually available in flight control. In this paper we use a state-space LTI model for the plant, different at each time step and corresponding to the parameter value α.

6. Designing the $MBPC$ controller

The controller design followed the general procedure used in conventional $MBPC$ cases.

This involved tuning the $MBPC$ controller for a plant model obtained by freezing the healthy quasi-LPV model at $\alpha = 0$. Note that this model is unstable. The plant model used for control had the following (approximated) continuous time transfer function:

$$\frac{n_Z}{\nu} = \frac{(s - 31.58)(s + 31.30)}{s(s + 6.0029)(s - 5.3984)}$$

A sampling time of $T_s = 0.01$s will provide a good approximation over the whole open-loop bandwidth.

Horizon dimensions were defined according with the general guide lines used in tuning $MBPC$ controllers [4]. The tuning was performed in the uncosntrained case.

The control horizon was chosen to be the same as the number of states ($N_u = 3$ in this case) which will ensure stability even in the worst case, namely when all the modes are unstable, should that ever occur [11].

The small sampling time has determined a long prediction horizon of $N_y = 30$ in order to predict 0.3s ahead. This was enough to ensure the performance criteria required [12].

The $MBPC$ control weighting matrix had to be made relatively small $R = 0.05$ due to the small effectiveness of the fin in this particular model.

The $MBPC$ tracking weighting matrix $Q = 1$ reflects the requirements on the time response characteristics (rise time, settling time and overshoot) for the closed loop system. These were obtained after some trial and error, starting with unit values for both weighting matrices.

The next step in design, after checking the unconstrained closed loop behaviour, was to enforce the constraints and check the constrained $MBPC$ performance. Once these steps were passed the controller was checked when performing adaptation with respect to α.

7. Prediction strategies

The main question raised while implementing the adaptive mechanism was which model should be used to provide predictions. Two strategies have been considered for use during manoeuvres which cause significant α variations:

- No *a priori* trajectory information. A single model was used across the whole prediction horizon (*ie* for $l = 1, \ldots, N_y$) but changed at each current time step (*ie* different for each k) in accordance with the measured α value.

- *A priori* trajectory information available. This allows the internal model to vary over the prediction horizon, but one needs to predict α over the prediction horizon in order to do this. It is important to base this prediction on the commanded trajectory, rather than the achieved one; this is more approximate, but it retains the QP structure of the optimisation problem. Basing it on the achieved trajectory would lose even the convexity property, so should be avoided if at all possible [1]. For the missile it is possible to obtain a prediction for α, based on the commanded trajectory for n_Z.

Note that these two strategies become the same in the special case when $N_y = N_u = 1$. In our evaluations we have so far simulated only the first prediction strategy.

It is advisable to take one of the above suggested paths. Otherwise, obtaining the models through the plant future dynamics, the resulting cost function is no longer quadratic in the command increment. This results in the standard $MBPC$ quadratic optimisation being transformed into a nonlinear constrained optimisation for which it is hard to guarantee a global solution [1].

8. Evaluation results

The two different capabilities of the controller – scheduling and reconfiguration – are investigated in the following simulations.

Firstly we show the behaviour of the closed loop with the different controllers when the system is subjected to a step of 0.1 in the non-dimensional controlled variable n_Z.

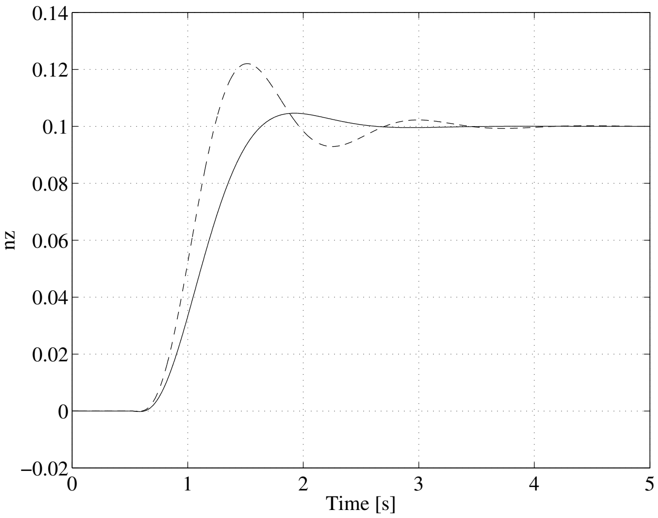

Figure 2: Comparison between the fixed and the adaptive $MBPC$ (adaptive–solid, fixed–broken)

The first controller is represented by the nominal design for which a fixed internal model, defined at $\alpha = 0$ is used, even though α varies during the manoeuvre.

In the second case the controller uses the fixed values for the cost function tuning parameters (N_y, N_u, Q, R), but the internal model is updated to follow the evolution of α, as described in Section 2.

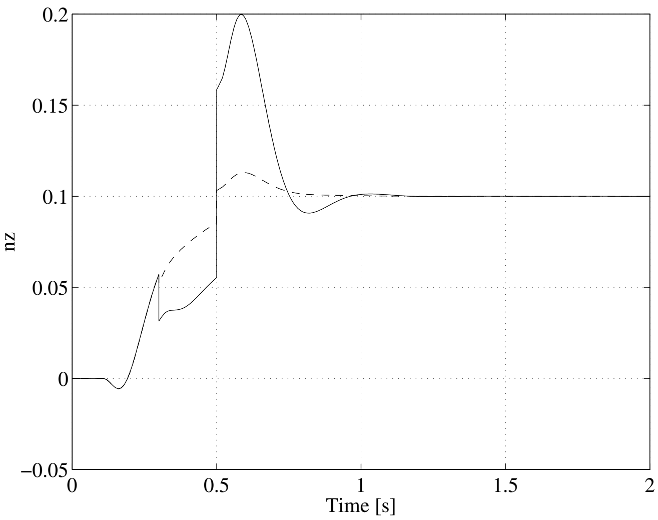

Figure 3: The reconfiguration capabilities provided by the new approach (10 % change–solid, 50 % change–broken)

For each such case we have ensured stability by an adequate choice of the control horizon (N_u greater than the system order). In this first case no failures are assumed for the controlled system.

The response of the system controlled by the adaptive $MBPC$ provides less overshoot and a shorter settling time as demonstrated in Figure 2.

In the second simulation we show the reconfiguration capability of the new approach. Successive structural failures of the missile were simulated: a failure on the body at $t = 0.3$s, and one on the fin at $t = 0.5$s. These failures modified the aerodynamic characteristics by 10% (broken line) and 50% (solid line), respectively. Figure 3 shows how the controller coped with each failure.

The failure was simulated by manipulating the nonlinear model dynamics. The impairment was represented in the high fidelity model by changing the coefficients relating the contributions of various aerodynamic surfaces to C_Z and C_m.

In both situations the new controller behaved well, providing scheduling and reconfiguration as required.

9. Conclusions

Our objective was to find a systematic way of reconfiguring control systems in the event of actuator and structural failures.

Assuming reliable FDI, the proposed approach gives:

- A systematic and compact way of representing high-fidelity models,
- Automatic updating of models following failure,
- Real-time production of internal models for $MBPC$ control,
- An autopilot structure which integrates the components shown in Figure 1.

The exploitation of the quasi-LPV models fits naturally in the $MBPC$ framework. Since no linearisation is involved these models are not limited to standard gain-scheduled designs.

Moreover, there is a wide class of systems exibiting output nonlinearities. They are particularly common in aerospace, but there are also process models which can be expressed in a similar form.

Although our approach is quite complex, we believe that the complexity is worthwhile if efective scheduling and reconfiguration can be achieved. The results of Section 8. are encouraging, and papers such as [10] indicate that others also take this view.

Acknowledgements

We would like to acknowledge Pembroke College Cambridge, Lundgren Fund, ORS scholarship, Cambridge Overseas Trust and CT Taylor Fund by which Mihai Huzmezan is supported. Present research is funded by DERA – Bedford.

References

[1] C.M. Chow, A.G. Kuznetsov, and D.W. Clarke. Using multiple model predictive control. In *Proceedings of the spring school "Adaptive and Predictive Control"*, Oxford, 21–22 Mar. 1996. Oxford University.

[2] G.D.Wood. *Control of Parameter–Dependent Mechanical Systems*. PhD thesis, University of Cambridge, 1995.

[3] D.E. Hoak. USAF Stability and Control DATCOM. Technical Report AFWAL-TR-83-3048, Air Force Wright Aeronautical Lab., Wright-Patterson AFB, 1978.

[4] M. Huzmezan and J.M. Maciejowski. *Robust Flight Control: A Design Challenge, GARTEUR*, volume 224 of *Lecture Notes in Control and Information Sciences*, chapter 12 – Predictive Control and 25 – Flight Management Using Predictive Control, pages 125–134 and 379–396. Springer-Verlag, 1997.

[5] R.A. Hyde. *H_∞ Aerospace Control Design*. Springer–Verlag, 1995.

[6] J.M. Maciejowski. Reconfigurable Control Using Constrained Optimisation. In *Proceedings of ECC'97*, Bruxelles, Belgium, 1997.

[7] A.S. Morse. Supervisory control of families of linear set point controllers. In *Proceedings of the 32nd Conference on Decision and Control*, San Antonio, Texas, Dec 1993. IEEE.

[8] K.S. Narendra and J. Balakrishnan. Adaptive Control Using Multiple Models. *IEEE Transactions on Automatic Control*, 42(2):171–187, 1997.

[9] M. Patcher, P.R. Chandler, and M. Mears. Reconfigurable Tracking Control with Saturation. *Journal of Guidance Control and Dynamics*, 18(5):1016–1022, Sep.–Oct. 1995.

[10] M. Patcher, P.R. Chandler, and L. Smith. Maneuvering Flight Control. In *Proceedings of the American Control Conference*, July 1997.

[11] J.B. Rawlings and K.R. Muske. The stability of constrained receding horizon control. *IEEE Transactions on Automatic Control*, 1993.

[12] J.S. Shamma and J.R. Cloutier. Gain-Scheduled Missile Autopilot Design Using Linear Parameter Varying Transformations. *Journal of Guidance Control and Dynamics*, 16(2):256–263, Mar.–Apr. 1993.