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

Fault-tolerant control using dynamic inversion and model-predictive control applied to an aerospace benchmark ⋆

D.A. Joosten * **T.J.J. van den Boom** ** **T.J.J. Lombaerts** ***

* *Delft Center for Systems and Control,Delft University of Technology, The Netherlands. (e-mail:* `d.a.joosten@tudelft.nl`*)*
** *Delft Center for Systems and Control, Delft University of Technology, The Netherlands. (e-mail* `a.j.j.vandenboom@tudelft.nl`*)*
*** *Control and Simulation division, faculty of Aerospace Engineering, Delft University of Technology, The Netherlands. e-mail:* `t.j.j.lombaerts@tudelft.nl`

Abstract: This paper features the combination of model-based predictive control and dynamic inversion into a constrained and globally valid control method for fault-tolerant flight-control purposes. The fact that the approach is both constrained and model-based creates the possibility to incorporate additional constraints, or even a new model, in case of a failure. Both of these properties lead to the fault-tolerant qualities of the method. Efficient distribution of the desired control moves over the control effectors creates the possibility to separate the input allocation problem from the inversion loop when redundant actuators are available. An important part of this paper consists of the application of the proposed theory to an aerospace benchmark of high complexity. It is shown through an example that the theory is well-suited to the task, provided that fault-detection and isolation information is available continuously.

Keywords: predictive control, fault-tolerant systems, inversion, aerospace control

1. INTRODUCTION

Physical damage to an aircraft, like the loss of a vertical tail, or the blockage of a control surface can lead to loss of controllability and/or stability. Such failures create a very challenging situation for the pilots, since such failures make it vastly more difficult, sometimes even impossible, to pilot the aircraft.

An example of such a situation, that was not survived by the crew, is the disaster that involved the El Al Boeing 747 freighter aircraft that crashed in the Bijlmermeer, near Amsterdam, The Netherlands, in 1992. In this particular case, separation of the two right wing engines caused significant loss of controllability and, next to that, structural changes, that eventually led to the crash. A simulation study by Smaili [1997] has shown that this failure was likely to be survivable, given the correct control inputs and a wisely chosen trajectory. This example, and many others, have clearly indicated that it is desirable to develop mechanisms that can assist the crew in such extraordinary situations.

The introduction of fly-by-wire systems has created the possibility to redistribute control effort over the actuators in an automated fashion. It has been the increase of computational power of such flight control systems that now allows for the investigation of fault-tolerant (FTFC) techniques. An overview of control methods that can be applied for FTFC purposes, has been compiled by Jones [2002]. The latter reference makes a clear distinction between passive and active methods. Passive methods are designed to accommodate failures through control design that is robust with respect to a set of system failures that is defined a priori. Active methods, on the other hand, assume that a fault detection and identification (FDI) is available that provides online failure information such that the FTFC controller can be adapted online. One active control method that is deemed very suitable for FTFC is model-predictive control.

In this paper the fault tolerant flight control (FTFC) problem is tackled using a combination of model predictive control (MPC) and feedback linearisation (FBL). Here, we present an approach that uses all the available freedom in the constrained inputs and that solves the problem of control allocation. The proposed theory builds upon previous work by the authors (Joosten et al. [2007]). The key contribution of this paper is the fact that the theory is applied to the an aerospace benchmark of high complexity.

This paper is organised as follows: Section 2 gives an overview of the different elements of the fault tolerant flight control setup and additionally shows how they interconnect in such a manner. The subsequent sections, subsections 2.1 through 2.3, discuss the implementation and inherent properties of these elements, which are feedback linearisation, model predictive control, constraint handling

⋆ This work was supported by the Dutch Technology Foundation (STW) under project number DMR6515.

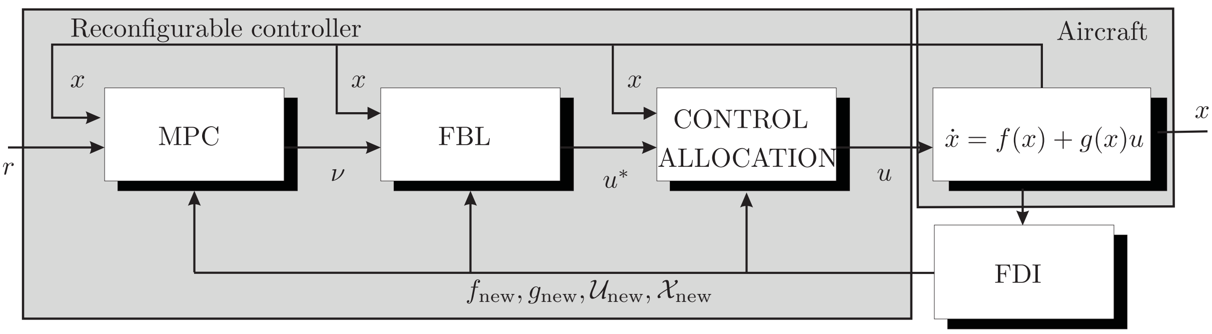

Fig. 1. Overview of the complete FTFC loop and the individual components. Additionally, the FDI block is shown to stress the importance of a failure detection method that delivers a new system description and a new set of constraints after the introduction of a failure.

and control allocation, respectively. Section 3 introduces the benchmark model and section 4 gives an example of the application of the FTFC method to this aircraft. The paper is concluded with an overview of the most important conclusions that can be drawn from the application of the theory to the aircraft model.

2. FAULT TOLERANT FLIGHT CONTROL SETUP

This section presents the reconfigurable control method that the authors propose. It is this reconfigurable method which is to be applied in a fault-tolerant setting. The synthesis of two different control techniques, model predictive control (MPC) and feedback linearisation (FBL), is what forms the basis of the design.

The first control method that is applied is MPC. MPC is selected because of its ability to cope with constraints on the input, state, or output of the system. Furthermore, since this is a discrete-time control method, it is possible to change the model in between the discrete time steps. The fact that MPC is model-based, and that it is possible to cope with constraints, makes it a serious candidate for FTFC of systems that feature an FDI mechanism. Both aspects determine the reconfigurable properties of the method. An example that considers the use of MPC in a simulation of the Bijlmermeer accident scenario is to be found in Maciejowski and Jones [2003].

MPC methods rely upon optimisation and require linear models in order for the solution of the optimisation problem to converge to a solution that is a global minimum. Feedback linearisation offers the possibility to find a nonlinear feedback law, such that given certain assumptions, the closed loop of the system and FBL controller has linear and time-invariant behaviour. An example of the combination of MPC and FBL in order to obtain globally valid and constrained control over the entry flight of a reentry vehicle is to be found in van Soest et al. [2006], and the combination of robust MPC and feedback linearisation is evaluated by van den Boom [1997]. Figure 1 provides an overview of how MPC and FBL are to be combined here.

The concept of a combination between FBL and MPC as to form a reconfigurable, globally valid, nonlinear, and constrained controller seems intuitive, but there are several interconnection issues that require attention. Such issues are caused by the fact that the number of system inputs is in general much larger than the number of states that are to be controlled, which is actually a prerequisite for FTFC. Furthermore, it is not a priori clear how the constraints on the inputs relate to the constraints of the MPC controller.

This section provides the theory of both MPC and FBL (or dynamic inversion, as we implement it) and discusses in-depth the interconnection issues that arise from this combination. Subsection 2.1 introduces the model structure and feedback linearisation. The next subsection provides the details of the MPC strategy that has been applied. Finally, subsection 2.3 provides details on how to distribute the desired control effort over the physical inputs.

For reasons of clarity, several assumptions, mainly because of simplicity, are posed here: it is assumed that online FDI information and hardware redundancy are available (i.e. the number of control effectors is greater than the number of controlled states), the model is assumed to be affine in the input and full state measurement or state-reconstruction is assumed to be available.

2.1 Feedback linearisation and dynamic inversion

Feedback linearisation is a control method that will obtain linear and decoupled input-output behaviour through application of a static and nonlinear feedback law. Aspects like relative degree, partial feedback linearisation and uncontrollable internal dynamics are important issues within the standard framework of feedback linearisation as presented by Isidori [1995], Slotine and Li [1991]. Feedback linearisation in its most basic form, input-state linearisation, is what is applied here. Input-state linearisation largely avoids the aforementioned issues. The presented implementation applies the concept of a virtual input and hence allows for the use of the available control effector redundancy in a further step, whereas FBL in its purest form does not.

This section starts with an introduction of the system-type that is considered here and continues to present the aspects that are involved in the combination of feedback linearisation and model predictive control. In this paper we consider nonlinear discrete-time systems, that are affine in the input, like

$$x(k+1) = f(x(k)) + g(x(k))u(k) \tag{1}$$

$$y(k) = h(x(k)) \tag{2}$$

where $x(k) \in \mathbb{R}^n$ is the state vector, where $u(k) \in \mathbb{R}^m$ is the vector of inputs, and where k indicates that this system is a discrete-time system with sampling-interval T. Furthermore, $f(x) \in \mathbb{R}^{n\times 1}$, $g(x) \in \mathbb{R}^{n\times m}$. Both the input $u \in \mathcal{U}$ and $x \in \mathcal{X}$ belong to a polyhedral set, i.e. they can be written as

$$\mathcal{U} = \{u \in \mathbb{R}^m |\, A\,u \le b\} \tag{3}$$

$$\mathcal{X} = \{x \in \mathbb{R}^n |\, A_x\,x \le b_x\} \tag{4}$$

for some matrices A, A_x and vectors b, b_x.

It is clear to see that, in order to invert the nonlinear dynamics, a choice of

$$g(x(k))u(k) = -f(x(k)) + \nu(k) \tag{5}$$

will result in decoupled closed-loop behaviour that equals

$$x(k+1) = \nu(k) \tag{6}$$

where $\nu(k) \in \mathbb{R}^n$ is a new input to the inverted system. The latter equation shows that the chosen control law decouples the system, such that the closed-loop constitutes a series of integrators in parallel. Furthermore, it is clear to see that when the number of inputs m is smaller than the number of states n, it will be impossible to invert the entire dynamics. When $m = n$ there will exist a unique solution to Equation (5) and when $m > n$ then there will exist a whole set of solutions $u(k)$ to this equation. It is necessary to make the remark that, that it is assumed in this paper that $m > n$, and hence input redundancy exists. Therefore, the input $u(k)$ will have to be allocated at every discrete-time step. The latter is commonly called nonlinear dynamic inversion (NDI), instead of FBL.

In summary, the input-state linearisation that is presented in this section leads to LTI behaviour that relates $\nu(k)$ to $x(k)$, and retains freedom in the allocation of $u(k)$. A restrictive result of the above is that the original input constraints on $u(k)$ must now be mapped into constraints on ν, since $\nu(k)$ will be controlled using model predictive control (see Figure 1). The next section will introduce an MPC algorithm that has been tailored to this situation, such that this issue can be avoided to a large degree.

Remark: It must be noted that discretisation of nonlinear dynamic systems is not at all trivial. In this paper the nonlinear system of the application example is sampled with sampling interval T and first order Euler integration is applied. The difference equation (1) is obtained from the original nonlinear system as follows

$$\dot{x} = f(x) + g(x)u \approx \frac{x(k+1) - x(k)}{T} \tag{7}$$

$$\Leftrightarrow$$

$$x(k+1) \approx Tf(x(k)) + x(k) + Tg(x(k))u \tag{8}$$

2.2 Model predictive control

Now that a linear discrete-time system (6) has been obtained through NDI, it is straightforward to apply model predictive control (MPC). MPC applies an internal model of the system under consideration. It is this model that is used to predict future values of dependent variables as a function of independent variables, in most cases the system input, over a prediction horizon. Application of a cost-function allows for the minimisation of this cost function over the horizon, subject to constraints. The first input is applied to the system and the optimisation is repeated during the next time-step.

A possible objective function, where the prediction horizon is chosen equal to N time steps, equals

$$J(\nu_k) = \sum_{i=1}^{N} e(k+i|k)^T Q e(k+i|k) \tag{9}$$

where $e(k+i|k) = \hat{x}(k+i|k) - x_r(k+i|k)$, and where $\hat{x}(k+i|k)$ is the predicted value of $x(k+i)$ at time k. $r(k) \in \mathbb{R}^n$ is the reference signal and $Q \succeq 0$ is a state weighting matrix, respectively.

If we introduce the following variables

$$\tilde{x} = \begin{bmatrix} x(k+1|k) \\ x(k+2|k) \\ \vdots \\ x(k+N|k) \end{bmatrix}, \quad \tilde{x}_r = \begin{bmatrix} x_r(k+1|k) \\ x_r(k+2|k) \\ \vdots \\ x_r(k+N|k) \end{bmatrix}$$

$$\tilde{u} = \begin{bmatrix} u(k|k) \\ u(k+1|k) \\ \vdots \\ u(k+N-1|k) \end{bmatrix}, \quad \tilde{\nu} = \begin{bmatrix} \nu(k|k) \\ \nu(k+1|k)_r \\ \vdots \\ \nu(k+N-1|k)_r \end{bmatrix}$$

and

$$\tilde{Q} = I_N \otimes Q$$

where I_N is an identity matrix of size N, and where the operator $\otimes$ indicates the Kronecker product of two matrices. The Kronecker product of two matrices A and B is defined as

$$A \otimes B = \begin{bmatrix} a_{11}B & \dots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \dots & a_{mn}B \end{bmatrix} \tag{10}$$

where a_{ij} is the i,j-th entry of matrix $A \in \mathbb{R}^{m\times n}$. Now, using relationship (6) the above objective function (9) can be expanded into

$$\begin{aligned} J(\tilde{\nu}, \tilde{x}_r) &= (\tilde{x} - \tilde{x}_r)^T \tilde{Q} (\tilde{x} - \tilde{x}_r) \\ &= (\tilde{\nu} - \tilde{x}_r)^T \tilde{Q} (\tilde{\nu} - \tilde{x}_r) \\ &= \tilde{\nu}^T \tilde{Q} \tilde{\nu} - 2\tilde{x}_r^T \tilde{Q} \tilde{\nu} \end{aligned} \tag{11}$$

The minimisation of $J(\tilde{\nu}, \tilde{x}_r)$ constitutes a quadratic programming problem (QP). The argument of the minimisation of this QP is the vector $\tilde{\nu}^*(k)$.

In order to be able to take into account the constraints on the physical input $u(k)$ it is necessary to incorporate Equation (5) which denotes the relationship between $\nu(k)$ and $u(k)$ and the constraints on input $u(k)$ as in (3). Both of these can be expanded over the horizon as follows

$$\underbrace{\begin{bmatrix} g(x(k)) & 0 \dots & 0 \\ \vdots & \vdots \ddots & \vdots \\ 0 & 0 \dots & g(x(k+N-1)) \end{bmatrix}}_{=\tilde{C}(x)} \tilde{u}(k) = \underbrace{\begin{bmatrix} -f(x(k)) \\ -f(x(k+1)) \\ \vdots \\ -f(x(k+N-1)) \end{bmatrix}}_{=\tilde{b}_{eq}(x)} + \tilde{\nu}(k) \quad (12)$$

and

$$\underbrace{(I_N \otimes A)}_{=\tilde{A}} \tilde{u}(k) \leq \underbrace{[1\ 1\ \dots\ 1]^T \otimes b}_{=\tilde{b}} \quad (13)$$

Hence it can be concluded that the optimization of cost-function (11) subject to (12) and (13) will produce the optimal vector $\tilde{\nu}^*(k)$. It must be noted, however, that $\tilde{u}(k)$ appears in the equality constraint (12) and that the same constraint also depends nonlinearly on the state $\tilde{x}(k)$. $\tilde{u}(k)$ is an independent variable and therefore it is necessary to append it to the cost-function (11) such that the constraints can also be incorporated in to the problem as follows

$$\min_{\tilde{\nu},\tilde{u}} \begin{bmatrix} \tilde{u} \\ \tilde{\nu} \end{bmatrix}^T \begin{bmatrix} 0 & 0 \\ 0 & \tilde{Q} \end{bmatrix} \begin{bmatrix} \tilde{u} \\ \tilde{\nu} \end{bmatrix} + \begin{bmatrix} 0 \\ -2\tilde{x}_r^T \tilde{Q} \end{bmatrix} \begin{bmatrix} \tilde{u} \\ \tilde{\nu} \end{bmatrix} \quad (14)$$

$$\text{s.t. } \left[\tilde{C} \mid -I_{Nn}\right] \begin{bmatrix} \tilde{u} \\ \tilde{\nu} \end{bmatrix} = \tilde{b}_{eq} \quad (15)$$

$$\left[\tilde{A}\ 0\right] \begin{bmatrix} \tilde{u} \\ \tilde{\nu} \end{bmatrix} \leq \tilde{b} \quad (16)$$

The minimisation of (14), s.t. (15) and (13) leads to a feasible $\tilde{u}^*$ and an optimal $\tilde{\nu}^*$. The latter may be interpreted as if the dynamic inversion were embedded into the MPC problem. It must be noted, however, that it is not possible to weight the input $\tilde{u}(k)$ during this phase because that impairs the state-tracking capability of the controller. The argument of the optimisation $\tilde{u}^*$ is not unique, since $g(x(k))$ is a wide matrix. Hence, it is possible to pose a second optimisation problem posed as a control allocation problem, which will be the subject of the next section.

One issue, that was already mentioned in the previous paragraph, is that the equality constraint (15) depends on the state in a nonlinear fashion. This constraint therefore has to approximated such that it is either constant or linearly dependent of the state at time k. Several solutions exist for these approximations, of which some are provided hereafter:

(1) assume that that $x(k)$ is constant over the horizon such that

$$\tilde{C} \approx I_n \otimes g(x(k)), \tilde{b}_{eq} \approx [1\ 1\ \dots\ 1]^T f(x(k));$$

(2) apply the input that was computed for the previous time-step to predict the evolution of the state over the horizon;
(3) assume that the system state will follow the reference state according to a stable and linear time-invariant (LTI) reference system;
(4) exploit a Jacobian linearization of $f(x(k))$ and $g(x(k))$ to obtain a local LTI model that can be applied to predict the evolution of the state over the horizon.

The authors acknowledge that what is presented in this section is a tailor-made MPC implementation, and refer to Maciejowski [2002] for an in-depth investigation of MPC and its properties in general.

Remark: The addition of $\tilde{u}(k)$ in (14) may seem redundant, but allows to avoid the complex and computationally intensive mapping of the polytope $\mathcal{U}$ that bounds $u(k)$ to a polytope that bounds $\nu(k)$ via the relationship

$$g(x(k))u(k) = -f(x(k)) + \nu(k) \quad (17)$$

Doing so, must be done every time-step and is very closely related to the subject of computational geometry. It is however well-known that projection methods, as are described in Preparata and Shamos [1985],are computationally very intensive and therefore not suitable for this application. Even the more advanced and much faster methods like the equality set projection algorithm by Jones et al. [2004] was shown to be prohibitive where computational complexity is concerned.

2.3 Control allocation

The previous sections have shown that it is possible to construct a globally valid, but constrained and nonlinear controller by means of a combination of MPC and FBL. Until now, however, we have only computed a feasible input u_k^*, in the previous section. This input is not unique, since in general the number of inputs is known to be larger than the number of controlled states. In many cases it will be desirable to be able to redistribute this feasible input such that, for instance, the absolute size of the inputs is minimal, or such that the change of the input with respect to the previous time-step is minimised.

Hence, since $m \geq n$ there is freedom in choosing u. One way to solve this problem involves the following quadratic programming problem

$$\begin{aligned} \min_u \ & u^T Q_u u + \Delta u^T R_u \Delta u \quad (18) \\ \text{s.t. } & g(x(k))u(k) = g(x(k))u^*(k) \\ & Au \leq b \end{aligned}$$

where $\Delta u = u(k) - u(k-1)$ and where $Q_u, R_u \succeq 0$ are input weighting matrices.

The above optimisation problem may be interpreted as follows: given one feasible input $u^*(k)$ that results from the MPC step, this control allocation problem will find a $u(k)$ that satisfies the mixed objective posed above: minimisation of the inputs and minimisation of the change of $u(k)$ with respect to the previous time-step, while satisfying the control allocation goal by means of the equality constraint $g(x(k))u(k) = g(x(k))u^*k)$.

It is this control allocation strategy that completes the FTFC setup that has been presented in this section, and the next section will show the merits of this FTFC method by means of an example that involves the nonlinear equations of motion of a fixed-wing aircraft.

3. INTRODUCTION TO THE GARTEUR AG16 BENCHMARK

This section provides a brief introduction to the Boeing 747-100/200 fault-tolerant control benchmark model to which the previously introduced control methodology will be applied. This model is used by the fault-tolerant control group (action group 16) of the Group for Aeronautical Research Europe (GARTEUR) in which several institutions from Europe participate.

The current version of the benchmark model is based on the Delft University aircraft simulation and analysis tool by van der Linden [1996] on the basis of which Smaili [1999] has implemented a Boeing 747-100/200 model for purposes of the reconstruction of the 1992 Bijlmermeer accident with an El-Al Boeing 747-100/200. Several contributors have since modified and improved the original model: Marcos [2001], Marcos and Balas [2003],Smaili et al. [2006].

In its current state the benchmark consists of a comprehensive model that can be evaluated in the *Matlab/SIMULINK* environment. The aerodynamic model and physical properties of this aircraft model correspond to actual wind-tunnel data of the aircraft-type under consideration (Hanke and Nordwall [1970]) and several failures, ranging in complexity from stuck actuators, to the complete scenario of the Bijlmermeer disaster.

The model comes with a complete reconstruction of the autopilot of this aircraft, but can also be flown in open-loop. In the open-loop setting, one has control over 30 different inputs, which represent the 25 different control surfaces, the 4 engines, and the landing gear. Integrated into the model are assessment criteria (Lombaerts et al. [2006]) for objective evaluation of the quality of the FTC methods of the different participants in GARTEUR AG16.

In the following section it will be shown that the control strategy proposed in this paper is suitable for retaining stability and tracking of a reference with the benchmark aircraft in case of a failure.

4. APPLICATION EXAMPLE

In this section we evaluate the performance of the combination of MPC and FBL as a reconfigurable control method. We do so in an example that involves a so-called stabiliser runaway. In this particular failure case the horizontal tailplane (stabiliser) of the aircraft moves towards its extreme position.

Before the actual example is discussed, it is necessary to introduce an important prerequisite that is required when using the benchmark model in this setting. In section 2.1 it was assumed that the system structure corresponds to Equation (1). In practice, however, the structure of the benchmark model is much more complex and, moreover, not affine in the input. Therefore, a recursive and online identification approach is applied in order to determine the aerodynamic parameters C_* in the proposed model structure

$$x(k+1) = f(C_*, x(k)) + g(C_*, x(k))u(k) \qquad (19)$$

$$y(k) = h(x(k)) \qquad (20)$$

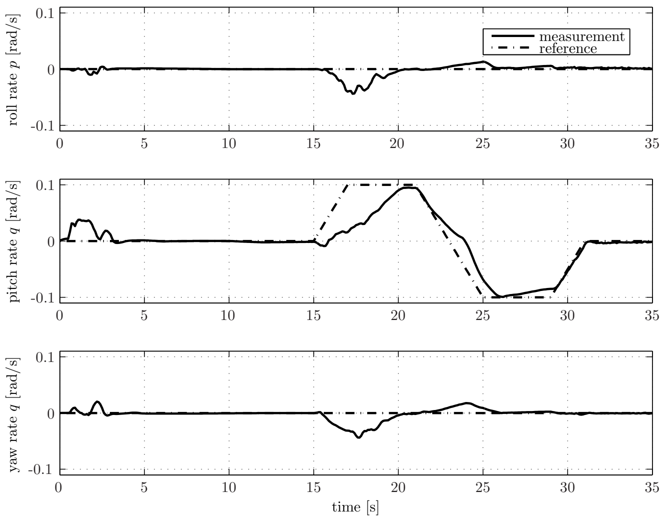

Fig. 2. Simulation result for the body rates p, q, r with respect to a reference after introduction of a stabiliser runaway fault at $t = 10$ [s]

These parameters are determined using the approach presented by Lombaerts et al. [2007]. Although not strictly required in the nominal case, the identification method is applied in both the nominal and the failure case here.

In this particular example, it is shown that a combination of the reconfigurable controller and the online identification algorithm can retain stability after introduction of the stabiliser runaway failure at time $t = 10$ [s]. At this time the stabiliser moves to its extreme trim angle of 2^o. Next to that, it is shown that, despite the stabiliser being inoperative and stuck at an extreme position, it is still possible to track a doublet-like reference signal with the pitch rate q [rad/s] using another combination of the control surfaces.

The states that are controlled, are the roll-rate p, the pitch rate q and the yaw rate r, respectively. The inputs that are used in this example are the 4 different aileron surfaces, the 4 elevator surfaces, the two rudder surfaces, and the stabiliser trim angle. The other inputs remain at their trim value for the initial condition of each simulation.

Figure 2 depicts the results that were obtained in simulation. Several important notions can be derived from this figure. First of all it can be seen from the figure that, although the online identification is initialised with data that was obtained off-line, it takes approximately 3 [s] for the closed loop to stabilise the system for the reference state $p, q, r = 0$. Furthermore, it clear to see, that although a failure is introduced at $t = 10$ [s] virtually no effect is noticeable in the state-response. The latter indicates that the controller successfully succeeds at redistributing the desired control effort over the remaining control surfaces while MPC automatically satisfies the physical constraints on the control surface deflections. And finally, it is easily seen from the figure that in spite of the failure of the stabiliser, it is still possible to track a reference with the pitch rate. It is assumed that extensive tuning of parameters like the state- and input weighting matrices Q, Q_u, R_u, the selected sampling interval T and the prediction horizon N will lead to greatly improved tracking behavior.

What remains to be said about this example is that the computational complexity of the control method is quite high. To be more precise: selection of an MPC prediction horizon $N \geq 2$ leads to a controller that cannot be evaluated in realtime, although this can be greatly improved upon through a more efficient implementation of the controller. Furthermore, although not visible in the provided results, the online identification algorithm suffers from lack of excitation when the system is controlled to be in steady-state for extended periods of time. Both of these issues are not addressed in this paper, but will be the topic of future research.

5. CONCLUSIONS AND FURTHER WORK

This paper has presented the combination of MPC and FBL into a constrained and globally valid control method and is as such an evolution of previous work (Joosten et al. [2007]). Using the proposed control method, it is possible to implement a reconfigurable flight control-law that is valid throughout the flight envelope. The reconfigurable properties are a result of efficient distribution of the desired control effort over the remaining and redundant control inputs. Furthermore, the method can take into account various input, state and output constraints. The latter is particularly useful when actuators get stuck in a certain position.

An example has been provided that shows that the combination of the proposed control strategy an online and recursive identification can both retain stability and track a reference when the body (pitch, roll, yaw) of the Boeing 747-100/200 benchmark model are controlled.

Practical issues that will be the topic of future research are related to the construction of a more computationally efficient variant of this controller. Additionally, it will have to be taken into account that the recursive identification scheme is applied in a closed-loop setting, whilst that fact is not explicitly accounted for at the moment.

From a theoretical point of view an interesting subject for future research is the addition of robustness to the FTFC method, whilst it is well-known that feedback linearisation and dynamic inversion methods are not particularly robust to modelling uncertainties. Such modelling uncertainties particularly arise in situations where FDI information is not available instantaneously. In order to achieve this, it is necessary to include theory for determination of the uncertainty in a model after feedback linearisation, like what is discussed by Juliana et al. [2005]. The same holds for the development of theory that explains the effect of discretisation on model uncertainty as to obtain an uncertain discrete-time feedback linearised system that is suitable for control with robust model predictive control methods like Kothare et al. [1996].

Increased robustness of the FTFC method will be of great importance in applications where there is latency in the FDI system. Robustness with respected to modelling uncertainty is required to guarantee stability until new and accurate FDI information becomes available after a failure has occurred.

ACKNOWLEDGEMENTS

This work was supported by the Dutch Technology Foundation (STW) under project number DMR6515.

REFERENCES

C.R. Hanke and D.R. Nordwall. The simulation of a jumbo jet transport aircraft. Technical Report D6-30643, National Aeronautics and Space Administration, September 1970. Volume II: Modeling Data.

A. Isidori. *Nonlinear control systems*. Springer, 1995.

C.N. Jones. Reconfigurable flight control. Technical report, Engineering Dept, University of Cambridge, 2002.

C.N. Jones, E.C. Kerigan, and J.M. Maciejowski. Equality set projection: A new algorithm for the projection of polytopes in halfspace representation. Technical Report CUED/F-INFENG/TR.463, Department of Engineering, University of Cambridge, 2004.

D.A. Joosten, T.J.J. van den Boom, and T.J.J. Lombaerts. Effective control allocation in fault-tolerant flight control with MPC and feedback linearization. In *Proceedings of the European Conference on Systems and control*, pages 3552–3559, Kos, Greece, July 2007.

S. Juliana, Q. Chu, J. Mulder, , and T. van Baten. The analytical derivation of nonlinear dynamic inversion control for parametric uncertain system. In *AIAA Guidance, Navigation, and Control Conference and Exnhibit*, number AIAA-2005-5849, San Francisco, CA, August 2005.

M.V. Kothare, V. Balakrishnan, and M. Morari. Robust constrained model predictive control using linear matrixn inequalities. *Automatica*, 32(10):1361–1379, 1996.

T. Lombaerts, D. Joosten, J. Breeman, H. Smaili, T van den Boom, Q. Chu, and J. Mulder. Assessment criteria as specifications for reconfiguring control. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, number AIAA 2006-6331, 2006.

T. Lombaerts, Q. Chu, J. Mulder, and D. Joosten. Real time damaged aircraft model identification for reconfiguring flight control. In *proceedings of the AIAA Atmospheric Flight Mechanics Conference and Exhibit*, number AIAA-2007-6717, 2007.

J.M. Maciejowski. *Predictive control: with constraints*. Harlow: Pearson Education, 2002. ISBN ISBN 0-201-39823-0.

J.M. Maciejowski and C.N. Jones. MPC fault-tolerant flight control case study: Flight 1862. *IFAC Safeprocess Conference*, 2003.

A. Marcos. A linear parameter varying model of the boeing 747-100/200 longitudinal motion. Master's thesis, Department of Aerospace and Engineering Mechanics, University of Minnesota, 2001.

A. Marcos and G.J. Balas. Boeing 747-100/200 aircraft fault tolerant and fault diagnostic benchmark. Technical Report AEM-UoM-2003-1, Department of Aerospace and Engineering Mechanics, University of Minnesota, 2003.

F.P. Preparata and M.I. Shamos. *Computational geometry: an introduction*. Springer, New York, 1985. ISBN 0-387-96131-3.

J.J.E. Slotine and W. Li. *Applied nonlinear control*. Prentice Hall Englewood Cliffs, NJ, 1991.

H. Smaili, J. Breeman, T. Lombaerts, and D. Joosten. A simulation benchmark for integrated fault tolerant flight control evaluation. In *AIAA Modeling and Simulation Technologies Conference and Exnhibit*, number AIAA-2006-6471, Keystone, CO, August 2006.

M.H. Smaili. Flight data reconstruction and simulation of El Al flight 1862. Master's thesis, Delft University of Technology, Delft, The Netherlands, 1997.

M.H. Smaili. Flightlab 747 benchmark for advanced flight control engineering v4.03. Technical report, Delft University of Technology, 1999.

T.J.J. van den Boom. Robust nonlinear predictive control using feedback linearization and linear matrix inequalities. In *Proceedings of the American Control Conference*, pages 3068–3072, June 1997.

C.A.A.M van der Linden. Dasmat delft university aircraft simulation model and analysis tool. Technical Report LR-781, Delft University of Technology, 1996.

W.R. van Soest, Q.P. Chu, and J.A. Mulder. Combined feedback linearization and constrained model predictive control for entry flight. *Journal of Guidance, Control and Dynamics*, 29 number 2:427–434, 2006.