Model Predictive Control Scheme for Rotorcraft Inverse Simulation

Giulio Avanzini*

Università del Salento, Brindisi, 72100, Italy

Douglas Thomson†

University of Glasgow, Glasgow, G12 8QQ, Scotland, UK

Alberto Torasso‡

Politecnico di Torino, Turin, 10129, Italy

A novel inverse simulation scheme is proposed for application to rotorcraft dynamic models. The algorithm is based on a model predictive control scheme that allows for a faster solution of the inverse simulation step, working on a lower–order, simplified helicopter model. The control action is then propagated forward in time on a more complete model. The algorithm compensates for discrepancies between the models by means of a simple guidance scheme.

The proposed approach allows for the assessment of handling quality potential on the basis of the most sophisticated model, adopted for the forward simulation, while keeping model complexity to a minimum level for the computationally more demanding inverse simulation algorithm. This allows for a faster solution of the inverse problem, if compared with the computational time necessary for solving the same problem on the basis of the full–order, more complex model. At the same time, the results are not affected by modeling approximations at the basis of the simplified one. The reported results, for an articulated blade, single main rotor helicopter model demonstrate the validity of the approach.

I. Introduction

Helicopter inverse simulation[1] has been an active topic of research since its first development over two decades ago with the seminal works of Thomson and Bradley[2] and Hess and Gao.[3] This technique in based on the determination of control inputs that allow a helicopter model to fly a specified manoeuvre. A wide plethora of methods for solving inverse simulation problems in flight mechanics has been considered in the past, which can be grouped into three major categories: (i) differential methods,[4] suitable for nominal problems only, where the number of control inputs equals that of the tracked variables; (ii) integration methods,[5] where the required control action is evaluated over a discrete time interval and can handle also redundant problems (e.g. by means of a local optimization approach[6]); and (iii) global methods,[7] where the time-history of the control variables is determined over the whole duration of the tracked manoeuvre by means of a variational approach.

As underlined in Ref. 1, the solution of the inverse problem is a task significantly more challenging for the rotorcraft case than for a conventional airplane, especially when individual blade dynamics is incorporated in the model.[8] Moreover, the issues related to the presence of transmission zeros and non-minimum phase response affect rotorcraft dynamics more seriously than fixed-wing aircraft models.

Among other methods, one of the advantages of integration methods is represented by their capability of dealing with complex, high order mathematical models of the vehicle on the basis of a solution scheme that can be applied with only minor variations to dynamical models of various order and complexity, provided that

*Professor, Faculty of Industrial Engineering, Edificio 14 Cittadella della Ricerca, S.S. 7 Km.7.3, AIAA Senior Member.

†Senior Lecturer, Department of Aerospace Engineering, James Watt South Building, AIAA Senior Member.

‡Ph.D. Student, Department of Aerospace Engineering, Corso Duca degli Abruzzi 24, AIAA Member.

the issue of unconstrained states is properly addressed. Although computation efficiency can be increased by application of a two-time-scale approach,[9] this remains a major concern in performing inverse simulation of complex high order models.

Bagiev et al.[10] have shown that a modification of the inverse simulation method to include a predictive step is able to provide more realistic solutions to the inverse simulation problem. Considering that the baseline algorithm may predict values which exceed the physical limits of the real vehicle, in this framework the predictive step is used in order to identify in advance such violations of physical limits or constraints, which might include mechanical limitations of control travel or control rates (based on hydraulic actuator stroke or other characteristics), limitations of rotor and tail rotor torque, and even structural limits of critical components.

Model Predictive Control may provide an answer to both the computation and the constraint violation problems. On one side the combined use of a low–order model to represent system behaviour together with the implementation of a receding horizon approach reduce the computational cost. At the same time the resulting algorithm allows for the identification and correction of trajectories which lead to constraint violations. In Model Predictive Control[19] the evaluation of the control law usually results from the solution of a finite horizon open-loop optimal control problem, using the current state of the plant as the initial state. The optimization yields an optimal control sequence and the first control in this sequence is applied to the plant which is then integrated until the next control step, when the same procedure is repeated. The control objective is usually to follow a user defined trajectory $\boldsymbol{y}(t) = \boldsymbol{y}_{des}(t)$ where $\boldsymbol{y}_{des}(t)$ is the desired evolution for the components of the vector of tracked outputs. The optimization problem is aimed at minimizing a stage cost based on the difference between real and desired output at any time step in the predictive space as well as on control activity and a terminal cost evaluated at the end of the integration (i.e. the receding horizon).

The algorithm proposed in this paper for the solution of inverse simulation problems uses a MPC scheme for the inverse simulation part of the routine. The complex model is substituted in the inverse simulation scheme by a lower–order model that requires a significantly smaller CPU time to solve the inverse problem (between 5 and 10 times faster). The architecture of the system allows for a continuous transmission of data on control activity and state position between the two models which limits the drift of the system output from the desired one. The scheme proposed in the paper is based on a special case of the optimization process, in particular on the solution of a nominal MPC problem (where the number of algebraic conditions matches that of control problem unknowns) enforced on the sole terminal cost (the output at the end of the integration step has to be equal to the desired one). At the present level the ability of the scheme to prevent violations of constraints has not yet been implemented.

The use of Model Predictive Control in the aerospace field is not new, as several control techniques based on prediction schemes have been proposed in literature[20–22] as a possible approach for the design of high performance controllers, often based on linear system. As a major contribution, the present research aims at the development of a nonlinear MPC step used for the precise and fast solution of an inverse simulation problem based on a complex individual blade helicopter model.

In what follows, the inverse simulation scheme will be presented in detail in the next section. Some manoeuvre examples are then proposed and discussed in the following paragraph, where two different lower–order, simplified models are used for the inverse simulation step, thus demonstrating that the algorithm can handle various degree of model complexity. A section of Conclusions ends the paper.

II. MPC scheme for IS

The proposed scheme allows the evaluation of the solution of an inverse simulation problem for a complex rotorcraft model, starting from the solution of the inverse simulation step obtained for a lower–order, simplified model. The complex model is used only in the forward simulation step, which is by far the computationally least demanding, while a great amount of time and computational burden can be saved by using a lower–order model in the inverse simulation step, which requires the numerical solution of a set of non-linear conditions on rotorcraft output at the final time of the discretization interval by means of an iterative procedure.

A. Models used for IS

The reference model used in the analysis is an individual blade representation of the UH60 based on the description by Howlett,[13] where fuselage is described by a full aerodynamic database of forces and moments depending on aerodynamic angles α_{fus} and β_{fus}. Blades dynamics include flap, lag and twist degrees of freedom. Main rotor inflow model is taken from Ref. 14. The reference model is represented by the 37-elements state vector $\boldsymbol{x} = (\boldsymbol{x}_B,\ \boldsymbol{x}_R,\ \boldsymbol{x}_{In})^T$ where $\boldsymbol{x}_B = (u, v, w, p, q, r, \phi, \theta, \psi)^T$ collects fuselage rigid body states, $\boldsymbol{x}_R = (\boldsymbol{x}_{R_1}^T, \boldsymbol{x}_{R_2}^T, ..., \boldsymbol{x}_{R_{N_b}}^T)^T$, with $\boldsymbol{x}_{R_i} = \left(\beta_i, \dot{\beta}_i, \zeta_i, \dot{\zeta}_i, \varphi_i, \dot{\varphi}_i\right)^T$, lists rotor flap, lag and twist angles and their derivatives, $\boldsymbol{x}_{In} = (\nu_0, \nu_{1c}, \nu_{1s}, \nu_{0_{TR}})^T$ represents main rotor and tail rotor inflow states.

The reference model is defined by a set of 37, time variant, nonlinear ordinary differential equations in the form

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}\left(\boldsymbol{x}, \boldsymbol{u}, \Psi(t)\right) \\ \boldsymbol{y} &= \boldsymbol{g}\left(\boldsymbol{x}\right) \end{aligned} \tag{1}$$

where the helicopter control vector $\boldsymbol{u} = (\theta_0, A_{1c}, A_{1s}, \theta_{0_{TR}})^T$ contains pilot commands on main rotor collective, lateral and longitudinal cyclic pitch and tail rotor collective pitch, whereas Ψ is rotor anomaly angle. For a constant rotor angular speed, it is $\Psi(t) = \Omega t$. Finally the reference model output vector $\boldsymbol{y}$ contains all the output variables needed for the guidance and inverse simulation steps.

The reference model, based on an individual blade approach, is inherently time variant and, in particular, oscillations in every state variable are expected at a frequency equal to (or multiple of) blade rotational speed, Ω, assumed constant in the sequel. As a consequence the trim conditions cannot be enforced in an algebraic way by setting to zero all states derivatives. A periodic trim[18] needs to be found by enforcing a periodicity condition on all the states in the form

$$\boldsymbol{x}\left(t\right) = \boldsymbol{x}\left(t + 2\pi/\Omega\right)$$

for a constant value of controls, $\boldsymbol{u}_0$. The values of control variables are chosen so as to determine (on average) a desired flight condition, defined in terms of airspeed, V, climb rate, $\dot{h}$, heading, χ (or turn rate, $\dot{\psi}$). The mean value of states over one rotor revolution

$$x_{i_0} = \frac{\Omega}{2\pi} \int_t^{t+2\pi/\Omega} x_i dt \tag{2}$$

is used as a reference for defining the state variables at trim. Several techniques can be found in the literature for solving the problem of helicopter periodic trim. In particular harmonic balance, periodic shooting, autopilot techniques have been proposed and compared,[17] also in the framework of helicopter performance evaluation.[11] In the present work, a periodic shooting approach derived from the work by McVicar and Bradley[18] is used. This technique has the advantage of being not particularly demanding for the considered model and at the same time it is flexible, being easily adapted to the evaluation of trim conditions for models of various levels of complexity.

The model used for the inverse simulation step is in general simpler than that used for forward simulation. In the present analysis the simplified model describes the main rotor with tip–path–plane dynamics and linear aerodynamics, which allow to analytically derive average rotor loads on fuselage, as in the work by Talbot et al.[16] The resulting 19 elements of the state vector $\tilde{\boldsymbol{x}}$ of the simplified model can be partitioned as in the previous case in the form $\tilde{\boldsymbol{x}} = (\tilde{\boldsymbol{x}}_B,\ \tilde{\boldsymbol{x}}_R,\ \tilde{\boldsymbol{x}}_{In})^T$, with the same fuselage rigid body and the inflow states, $\tilde{\boldsymbol{x}}_F$ and $\tilde{\boldsymbol{x}}_{In}$, respectively, while rotor is represented by tip–path–plane second order dynamics, $\tilde{\boldsymbol{x}}_R = \left(\tilde{\beta}_0, \dot{\tilde{\beta}}_0, \tilde{\beta}_{1c}, \dot{\tilde{\beta}}_{1c}, \tilde{\beta}_{1s}, \dot{\tilde{\beta}}_{1s}\right)^T$ where $\tilde{\beta}_0,\ \tilde{\beta}_{1c},\ \tilde{\beta}_{1s}$ are respectively coning, longitudinal and lateral flapping coefficients.

The dynamics of the simplified model is thus defined by means of a set of 19 nonlinear time–invariant ordinary differential equations, in the form

$$\begin{aligned} \dot{\tilde{\boldsymbol{x}}} &= \tilde{\boldsymbol{f}}\left(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{u}}\right) \\ \tilde{\boldsymbol{y}} &= \tilde{\boldsymbol{g}}\left(\tilde{\boldsymbol{x}}\right) \end{aligned} \tag{3}$$

where $\tilde{\boldsymbol{u}} = \left(\tilde{\theta}_0, \tilde{A}_{1c}, \tilde{A}_{1s}, \tilde{\theta}_{0_{TR}}\right)^T$ is the command vector and $\tilde{\boldsymbol{y}}$ is the output vector. Note that the states, commands and outputs of the model used for the inverse simulation step are defined by symbols with a

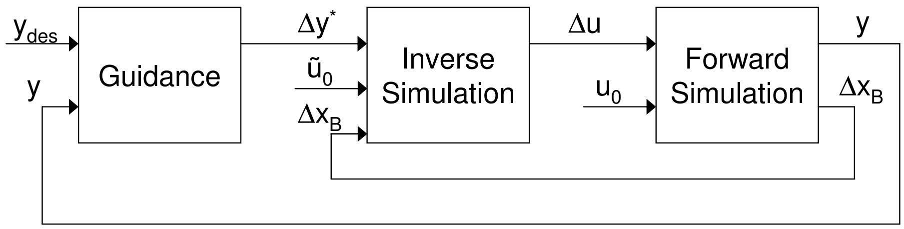

Figure 1. Description of the approach for the solution of the inverse simulation problem

tilde, in order to underline the fact that, in general, they may assume different values with respect to their counterparts in the reference models due to the difference in modelling level and tracking error of the output variables during the procedure. The time-variant reduced-order model can be trimmed by means of algebraic tools, simply enforcing the condition

$$\tilde{\boldsymbol{f}}\left(\tilde{\boldsymbol{x}}_0, \tilde{\boldsymbol{u}}_0\right) = 0$$

where $\tilde{\boldsymbol{x}}_0$ and $\tilde{\boldsymbol{u}}_0$ are the state and control variables at trim.

Since the two models may generate slightly different trim conditions, during the routine the variation of states and commands from their trim value is used rather than their actual values. Using this approach the difference between equilibrium states has no impact and causes no drift in the evaluation of the dynamic behaviour. The symbols $\Delta\boldsymbol{u} = \boldsymbol{u} - \boldsymbol{u}_0$ and $\Delta\tilde{\boldsymbol{u}} = \tilde{\boldsymbol{u}} - \tilde{\boldsymbol{u}}_0$ thus indicate control variable increments with respect to the considered reference trim condition. Similarly, for the state vector the following increments are defined $\Delta\boldsymbol{x}_R = \boldsymbol{x}_R - \boldsymbol{x}_{R_0}$, $\Delta\boldsymbol{x}_F = \boldsymbol{x}_F - \boldsymbol{x}_{F_0}$,$\Delta\boldsymbol{x}_{In} = \boldsymbol{x}_{In} - \boldsymbol{x}_{In_0}$ for rotor, fuselage and inflow states, respectively.

B. Inverse simulation algorithm

The approach for the solution of the inverse problem is described in Fig. 1. Three major blocks form the architecture of the algorithm. The forward simulation block performs the forward simulation of the reference model. The inverse simulation block is responsible for the evaluation of the command time–history that achieves the desired increment $\Delta\boldsymbol{y}^\star$ for the tracked output variables. Finally the guidance block provides the desired output to the inverse simulation block based on the desired trajectory and the actual output function of the reference system.

More in detail, at any time step t_k the inverse simulation bloc evaluates the control action $\tilde{\boldsymbol{u}}$ which is then passed to the forward simulation as command displacement from trim condition, assuming $\Delta\boldsymbol{u} = \Delta\tilde{\boldsymbol{u}}$. From the knowledge of the initial trim condition the forward simulator integrates the equations of motion for the reference model assuming a constant value of the control variables, $\boldsymbol{u} = \boldsymbol{u}_0 + \Delta\boldsymbol{u}$, over a time step equal to the inverse simulation step Δt.

Since the model used in the forward simulation is different from that used in the inverse simulation, both the states $\Delta\boldsymbol{x}\left(t_k + \Delta t\right)$ and output variables $\boldsymbol{y}\left(t_k + \Delta t\right)$ achieved at the end of the simulation step are (hopefully only slightly) different from their counterparts for the IS step, $\Delta\tilde{\boldsymbol{x}}\left(t_k + \Delta t\right)$ and $\tilde{\boldsymbol{y}}\left(t_k + \Delta t\right)$, determined on the basis of a simplified model. It is thus reasonable that the output variables $\boldsymbol{y}\left(t_k + \Delta t\right)$ are somewhat different from $\boldsymbol{y}_{des}\left(t_k + \Delta t\right)$, as enforced for the simpler model during the IS step.

Moreover, the matching condition between the outputs and their desired values is enforced at a time $t_k + T$, with $T > \Delta t$, where the receding time–horizon T provides a sufficient settling time for uncontrolled dynamics, as discussed in detail below and in Ref. 12. On the converse, the output of the reference system is taken at time $t_k + \Delta t$. As a consequence a guidance step is added to the inverse simulation process in order to limit the drift of the system from the desired trajectory, by updating the desired output variables as a function of the error exhibited by the reference model at the end of the forward simulation step. The desired output $\boldsymbol{y}_{des}$ for the following step $t_{k+1} = t_k + \Delta t$ is thus corrected as a function of the forward simulation output $\boldsymbol{y}\left(t_k + \Delta t\right)$ at the end of the k–th step using an approach later described in Eq. (9).

The IS problem is solved by means of an integration algorithm. In a standard inverse simulation approach,[12] once a desired variation with time of the output, $\boldsymbol{y}_{des}(t)$, is available (i.e. a manoeuvre profile like those required by ADS–33 specifications[15]) the helicopter equations of motion are integrated from an initial condition $\boldsymbol{x}_I = \boldsymbol{x}_k$ at time t_k over a time interval Δt (the inverse simulation time step) for a piece–wise constant value $\boldsymbol{u}_k^\star$ of the control variables. The resulting value $\boldsymbol{y}_F = \boldsymbol{g}(\boldsymbol{x}_F)$ of the output variables at time $t_F = t_{k+1} = t_k + \Delta t$ is therefore a function of the (given) initial state $\boldsymbol{x}_k$ and of the (unknown) constant control action, $\boldsymbol{u}_k^\star$, which is thus evaluated iteratively.

This approach is extremely demanding from the computational point of view especially for individual blade models featuring as many as 37 states, such as the helicopter model adopted for this study. The resulting computational time may become considerable high, also on modern CPUs. In order to reduce the computational burden, the inverse problem is here solved on the basis of a lower–order, simplified model. Some changes to the inverse simulation integration method are required, in order to achieve robustness and in general better performances.

As a first issue, since a reduced order model is adopted for the IS step, at any time t_k a proper set of initial conditions is required for integrating the set of ODE represented by Eq. (3). The ideal choice of setting $\tilde{\boldsymbol{x}}_I = \boldsymbol{x}_k$ is ruled out by the fact that the two vectors have a different number of components and, moreover, some of the states would not be accessible to direct measurements, if the algorithm is implemented as a MPC controller for an actual vehicle, rather than an off–line inverse simulation method for a complex helicopter model. For this reason, the issue of state initialization at the beginning of every integration step needs to be addressed especially for the inverse simulation block. For the forward simulation, the states at the beginning of the k–th step are simply given by the value assumed at the end of the previous one $\boldsymbol{x}_{I_k} = \boldsymbol{x}(t_k) = \boldsymbol{x}(t_{k-1} + \Delta t) = \boldsymbol{x}_{F_{k-1}}$.

For the inverse simulation step, on the converse, the initialization of states must rely at least partially on the knowledge of the states of the reference forward model in order to prevent a drift between the two models and consequent loss of control when implementing the control action derived from the simplified model on the full–order one. Two options are here considered. In the first case (technique A) as much information as possible is passed from the complete model to the reduced order one. Increments for fuselage and inflow variables are evaluated and the initial states for the inverse simulation step are given by

$$\tilde{\boldsymbol{x}}_B(t_k) = \tilde{\boldsymbol{x}}_{B_0} + [\boldsymbol{x}_B(t_k) - \boldsymbol{x}_{B_0}] \tag{4}$$

$$\tilde{\boldsymbol{x}}_{In}(t_k) = \tilde{\boldsymbol{x}}_{In_0} + [\boldsymbol{x}_{In}(t_k) - \boldsymbol{x}_{In_0}] \tag{5}$$

where $\boldsymbol{x}_{B_0}$ and $\tilde{\boldsymbol{x}}_{B_0}$ are the rigid body state trim conditions for the reference and inverse model respectively and $\boldsymbol{x}_B(t_k)$ is the reference model rigid body state at the end of the previous forward integration step. Similarly $\boldsymbol{x}_{In_0}$ and $\tilde{\boldsymbol{x}}_{In_0}$ represent the inflow states for the reference and inverse models, and $\boldsymbol{x}_{In}(t_k)$ is the reference model inflow state at the end of the previous forward integration step. As for rotor states, coning, longitudinal and lateral flapping coefficients at t_k are evaluated by means of multiblade coefficients:

$$\begin{aligned}
\beta_0(t_k) &= \frac{1}{N_{bl}} \sum_{j=1}^{N_{bl}} \beta_j(t_k) \\
\beta_s(t_k) &= \frac{2}{N_{bl}} \sum_{j=1}^{N_{bl}} \beta_j(t_k) \sin\psi_j \\
\beta_c(t_k) &= \frac{2}{N_{bl}} \sum_{j=1}^{N_{bl}} \beta_j(t_k) \cos\psi_j
\end{aligned} \tag{6}$$

where N_{bl} is the number of blades. Letting $\boldsymbol{\beta} = (\beta_0, \beta_s, \beta_c)^T$, the initial condition for rotor states is defined as

$$\tilde{\boldsymbol{x}}_R(t_k) = \tilde{\boldsymbol{x}}_{R_0} + [\boldsymbol{\beta}(t_k) - \boldsymbol{\beta}_0] \tag{7}$$

Technique B is based on the hypothesis that only rigid body states $\boldsymbol{x}_B$ of the reference model are truly observable as it would happen in a real-time application of the algorithm in the form of an actual MPC scheme. The same displacement of fuselage states from trim condition is enforced in the inverse model initial states as in Eq. (4), but inflow and rotor states are not updated. Inflow and rotor states are assumed as not observable and therefore they are initialized with the value at the end of the last inverse simulation run $\boldsymbol{x}_{R,In}(t_k) = \boldsymbol{x}_{R,In}(t_{k-1} + \Delta t)$.

The choice of selecting rigid body states only as observable states maintains a link to Model Predictive Control procedures. In fact, if a real system replaces the forward simulation model, only some states would be observable. In particular linear and angular velocities as well as attitude variables are usually available from GNC sensors and as a consequence they can be fed to the inverse model in the above mentioned routine. On the converse, rotor and inflow states are in general non-observable states and therefore no feedback of their actual value from the controlled plant can be provided to the inverse simulation model in a realistic scenario.

In both cases, simplified model states are integrated over a time interval $T = N\Delta t$, that is, a time interval longer than the inverse simulation step Δt, using a piece–wise constant value $\tilde{\boldsymbol{u}}_k^\star$ for the control variables. The longer integration time allows for some fast uncontrolled dynamics to settle down before the instant when the objective function is evaluated. The selection of the integration time results from a trade-off between computational time and stability of the method. In fact short integration time may excite uncontrolled dynamics and lead to an unstable or highly oscillatory response of the system, both of which should be discarded as poor and/or impractical solutions of the inverse problem. Furthermore this approach is common practice in Model Predictive Control, where the receding horizon used for the forward prediction of system behaviour and evaluation of control activity is usually 3 to 10 times the controller time step.

In the routine developed for the present work the integration time T is selected so that $T = N\Delta t$ with $N = 3$ for the results proposed in the following section. The resulting value $\tilde{\boldsymbol{y}}_F = \tilde{\boldsymbol{g}}(\tilde{\boldsymbol{x}}_F)$ of the output variables at time $t_F = t_k + T = t_k + N\Delta t$ is thus a function of the (given) initial state $\tilde{\boldsymbol{x}}_k$ and of the (unknown) constant control action, $\tilde{\boldsymbol{u}}_k^\star$.

Control variables can then be determined in such a way that $\tilde{\boldsymbol{y}}_F$ matches the value of $\boldsymbol{y}_{des}$ at time t_F, that is, the inverse problem can be stated in terms of a set of p algebraic equations in the form

$$\tilde{\boldsymbol{y}}_F = \tilde{\boldsymbol{F}}(\tilde{\boldsymbol{x}}_k, \tilde{\boldsymbol{u}}_k^\star) = \boldsymbol{y}_{des}(t_F) \tag{8}$$

with m unknowns. When $m = p$, the problem is nominal and, if well posed, it can be solved by means of standard numerical techniques, such as Newton–Raphson (NR) method. If $m > p$ the problem is redundant, as in many aeronautical applications for fixed and rotary–wing aircraft, when 4 controls are available for tracking 3 trajectory variables.

As a further variation to a standard integration method, a different definition of the algebraic system is adopted in this paper, where, rather than directly solving Eq. (8) in terms of the actual value of the tracked variables at time t_F, their increments over the time step between t_I and $t_F^\star$ are required to be equal. Equation (8) is thus replaced with

$$\begin{aligned} \Delta\tilde{\boldsymbol{y}}^\star &= \tilde{\boldsymbol{F}}(\tilde{\boldsymbol{x}}_k, \tilde{\boldsymbol{u}}_k^\star) - \boldsymbol{y}(t_k) = \\ &= \tilde{\boldsymbol{F}}(\tilde{\boldsymbol{x}}_k, \tilde{\boldsymbol{u}}_k^\star) - \boldsymbol{g}(\boldsymbol{x}_k) = \\ &= \boldsymbol{y}_{des}(t_F) - \boldsymbol{y}_{des}(t_I) + K\left[\boldsymbol{y}_{des}(t_I) - \boldsymbol{g}(\boldsymbol{x}_k)\right] \end{aligned} \tag{9}$$

where the additional term in square brackets multiplied by a gain K avoids that the actual solution "drifts" away from the desired path because of the incomplete implementation of the considered step during the forward propagation, as outlined above. This term also enforces asymptotic convergence on the tracked variables when they achieve a steady value.

If on one side the condition on the output function is enforced on the model used in the inverse simulation step described by Eq. (3), on the other hand the correction of the desired output due to the discrepancies of actual value of the output function is done independently in the Guidance step described in Fig. 1. Since the actual system output $\boldsymbol{y}(t_k)$ is available, the correction of the drift term is based on this value, as described in Eq. (9). By some simple manipulation, Eq. (9) can be rearranged as

$$\tilde{\boldsymbol{F}}(\tilde{\boldsymbol{x}}_k, \tilde{\boldsymbol{u}}_k^\star) = \boldsymbol{y}_{des}(t_F) + (K-1)\left[\boldsymbol{y}_{des}(t_I) - \boldsymbol{g}(\boldsymbol{x}_k)\right]$$

where for $K = 0$ the additional term disappears and one simply requires that the increment of the actual output variables at the end of the whole inverse simulation step $T = t_F - t_I$ equals the increment for the desired variation of $\boldsymbol{y}$.

As described above, in order for the inverse problem to be nominal and therefore to be solved via Newton–Raphson method, the number of algebraic equations has to match the number of controls (i.e. 4 for the present case). In the inverse simulation problem the helicopter must follow a desired trajectory. The flight

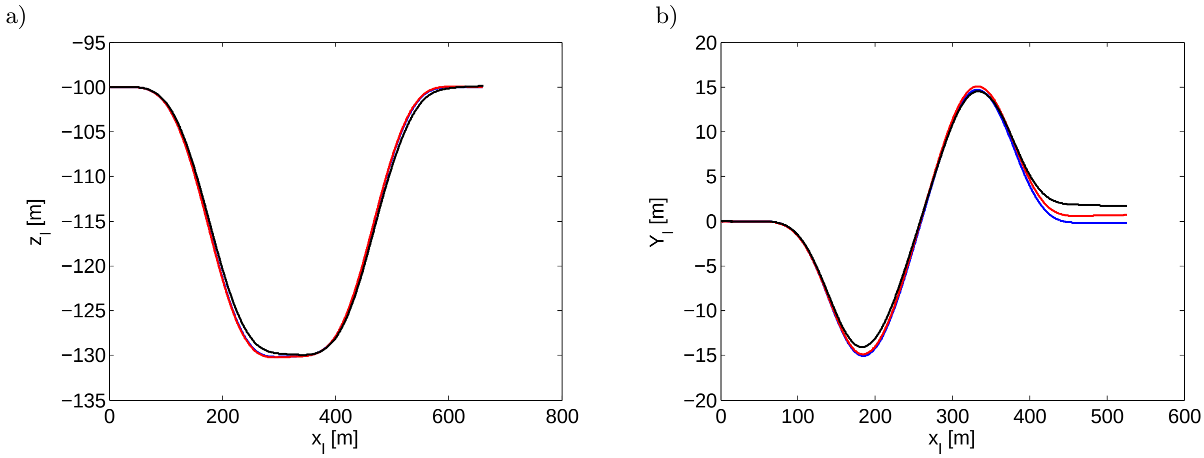

Figure 2. Maneuvers trajectories: (a) hurdle-hop; (b) slalom; – IS problem solved with reference model, – IS problem solved with inverse model, – IS problem solved with MPC approach.

task element can be enforced on the model by setting as objective function at every time step the inertial position, the inertial velocity components or the inertial acceleration. A choice between the possible objective functions is required, where setting the position as desired variables may lead to instability in the algorithm, while choosing the acceleration components may lead to large drift from the desired trajectory as the system integrates twice the error on the considered problem objective function. The inertial velocities were then chosen as the baseline desired output to be tracked by means of the inverse problem.

As a final issue, setting one of the previous conditions to follow the desired variation with time enforces only 3 conditions, when 4 control variables are available. A further objective function needs to be selected to make the problem nominal. In particular $\psi = 0$ is chosen in those cases when the helicopter is required to face a particular direction during the whole manoeuvre (e.g. hurdle-hop or lateral repositioning) whereas a condition of zero lateral acceleration in body reference frame ($a_y = 0$) is enforced in lateral-directional manoeuvres.

III. Results and Discussion

The approach described in the previous section is tested in this paragraph with a series of manoeuvres taken from ADS-33 E standard[15] or from inverse simulation literature.[1,9,12] In particular a longitudinal manoeuvre (a hurdle hop) and a lateral-directional one (a slalom) are used to show the effectiveness of the approach. In both cases, the trim conditions of the reference and the inverse simulation models need to be evaluated for initializing the procedure. Then, following the approach described in the previous section, the inverse solution is calculated using inertial velocities as desired output function. The nominal MPC problem is closed imposing the terminal conditions $\psi = 0$ in the longitudinal hurdle-hop and $a_y = 0$ in the slalom. Results produced by means of “full state feedback” from the complete to the simplified model at the end of the simulation step (technique A) almost perfectly match the inverse solution obtained from the reference model alone. For this reason only results produced by technique B (feedback on rigid body states only) will be presented in the sequel. These results are compared with the solution of the inverse simulation problem based on the same manoeuvres obtained by a more traditional inverse simulation method[12] where the forward and inverse simulation phases are based on the same model. Trajectory and command time histories are compared to the results of these two inverse simulation problems based on the complete and on the simplified models, respectively.

Figure 2 represents the trajectory realized by the three different approaches in the two manoeuvres. It is clear that in the hurdle-hop manoeuvre the three algorithm are able to follow the reference trajectory with great precision. In the slalom manoeuvre the performance of the MPC approach is still good even if a small drift from the desired trajectory emerges in the last segment.

The command time histories in the two manoeuvres show that the approach based on MPC is able to generate the correct command to the reference model, at a fraction of the computational cost if compared to

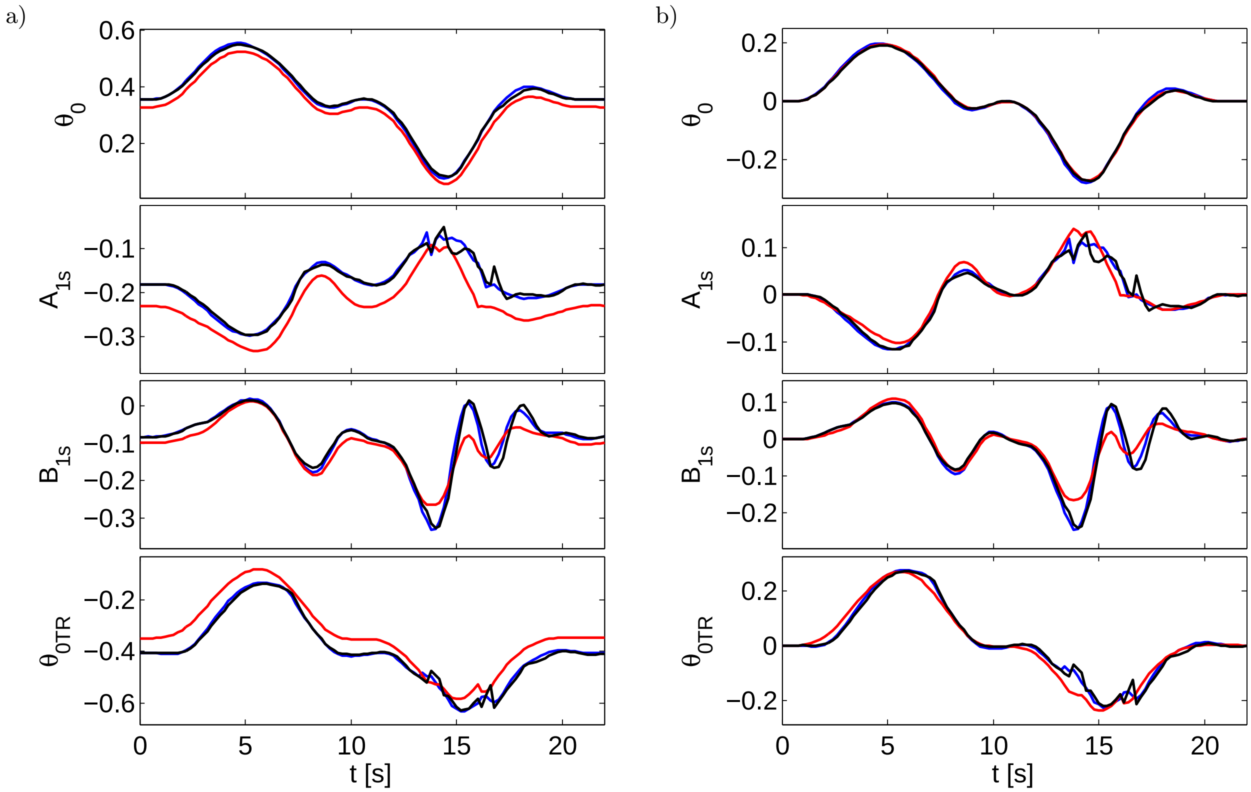

Figure 3. Command time-histories in hurdle-hop manoeuvre: (a) total command travel and (b) displacement from trim. – IS problem solved with reference model, – IS problem solved with inverse model, – IS problem solved with MPC approach.

the inverse solution of the full, individual blade model. Figure 3 compares the command time histories for the three approaches in a hurdle-hop manoeuvres. The commands are scaled with respect to their maximum travel. In particular main rotor collective θ_0 can vary between 0 and 1, while all other commands are scaled between minimum and maximum values set respectively at -1 and $+1$. Figure 3.a represents the command travel, while figure 3.b plots the displacement from command trim condition in order to remove the impact of differences in equilibrium conditions. The result of the inverse simulation problem based on the complete model is taken as a reference, represented in the figures by the blue line. Furthermore the MPC approach (black line) is expected to behave not worse than the IS approach based on the lower–order model (red line). It is clear that the MPC approach is able to represent very well the behaviour of the reference model. A small vibration is present for both the reference model and the MPC solution during the descent phase of the hurdle hop. This is a very critical phase of the manoeuvre due to the quick change of aerodynamic conditions around the rotor during a steep unsteady descent. Nevertheless, apart from this small "vibration", the MPC approach performs really well in predicting the complex model command required to perform the manoeuvre.

A similar behaviour can be seen in Fig. 4 where the results for a slalom manoeuvre are shown. In this case the results of the MPC approach are still satisfactory and, considering all command lines, better than the solution evaluated using only the lower–order model (red line). Nevertheless the predicting capability of the MPC approach is not as precise as in the previous manoeuvre. This may be due to the fact that the slalom manoeuvre couples the longitudinal and lateral variables of main rotor and inflow models. The level of rotor modelling is very different between the two models (individual blade with flap, lag and twist degrees of freedom compared to a simple 2^{nd} order tip path plane dynamics). As a consequence some differences in rotor response to such a complex aerodynamic environment appear as reasonable. Furthermore, as described above, there is no rotor state feedback from the reference model to the simpler one, so the inverse simulation and the forward one may start from quite different rotor and inflow initial conditions. This leads to the evaluation of a control $\Delta\boldsymbol{u}^\star$ which meets the output requirement for the lower order model, but fails to provide a very accurate estimate for the reference one. The introduction in the MPC approach of rotor and

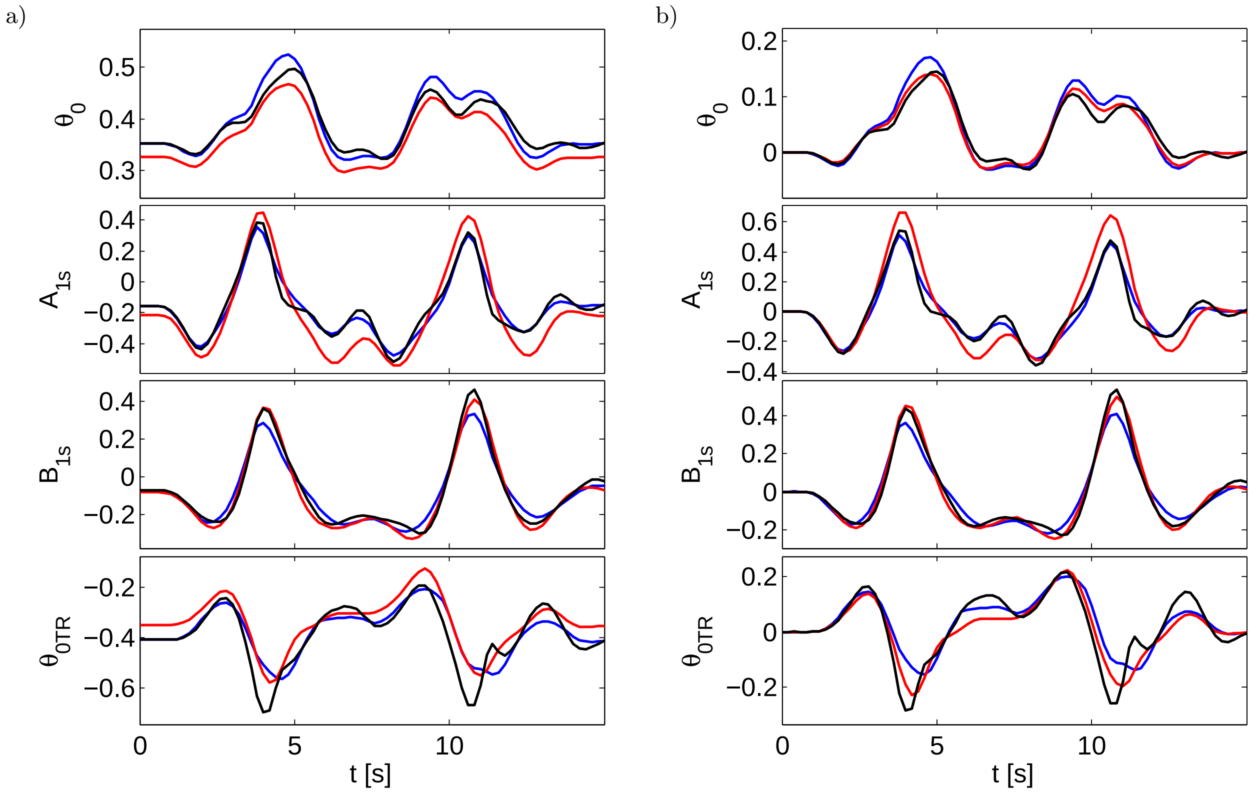

Figure 4. Command time-histories in slalom manoeuvre: (a) total command travel and (b) displacement from trim. – IS problem solved with reference model, – IS problem solved with inverse model, – IS problem solved with MPC approach.

inflow states feedback enables a more precise estimation of complex model commands.

Since the scheme works well with the lower–order model described in the second section, it has been tested with an even simpler model. This lower–order model is represented by system state $\hat{\boldsymbol{x}} = (\hat{\boldsymbol{x}}_B,\ \hat{\boldsymbol{x}}_R\ )^T$ where $\hat{\boldsymbol{x}}_B = \left(\hat{u}, \hat{v}, \hat{w}, \hat{p}, \hat{q}, \hat{r}, \hat{\phi}, \hat{\theta}, \hat{\psi}\right)^T$ represents fuselage states and $\hat{\boldsymbol{x}}_R = \left(\hat{\beta}_0, \hat{\beta}_{1c}, \hat{\beta}_{1s}\right)^T$ represents first–order dynamics rotor states (coning, longitudinal and lateral states). Further simplifying assumptions of this model include the inflow which is assumed uniform and quasi–steady and the fuselage aerodynamic description which is based on parasite drag area and not on complete aerodynamic database as in the previous models. Figure 5 collects the results of the MPC inverse simulation problem based on this minimum complexity model. It can be seen that even if the model used for the inverse simulation step is very simple, the results are still satisfactory.

From a computational point of view the solution of the classical inverse simulation problem as described in Ref. 12 with the lower–order model is 4 time faster than the reference complex model. Using the MPC approach the computational burden is almost identical to the inverse simulation problem based on the simple model as the forward simulation performed with the complex model absorbs just a marginal time in the whole process. As a consequence with the same computational burden (4 times faster than the traditional inverse simulation based entirely on the complex model) it is possible to evaluate almost exactly the command time history required to perform the manoeuvre. Using the minimum complexity rotorcraft model described before a further 20% computational time is saved.

IV. Conclusions and Future Work

A novel approach to the solution of inverse simulation problems based on a model predictive control scheme was proposed. The approach is able to significantly reduce the computational burden required by the inverse simulation of a complex nonlinear helicopter model by using a lower–order model in the

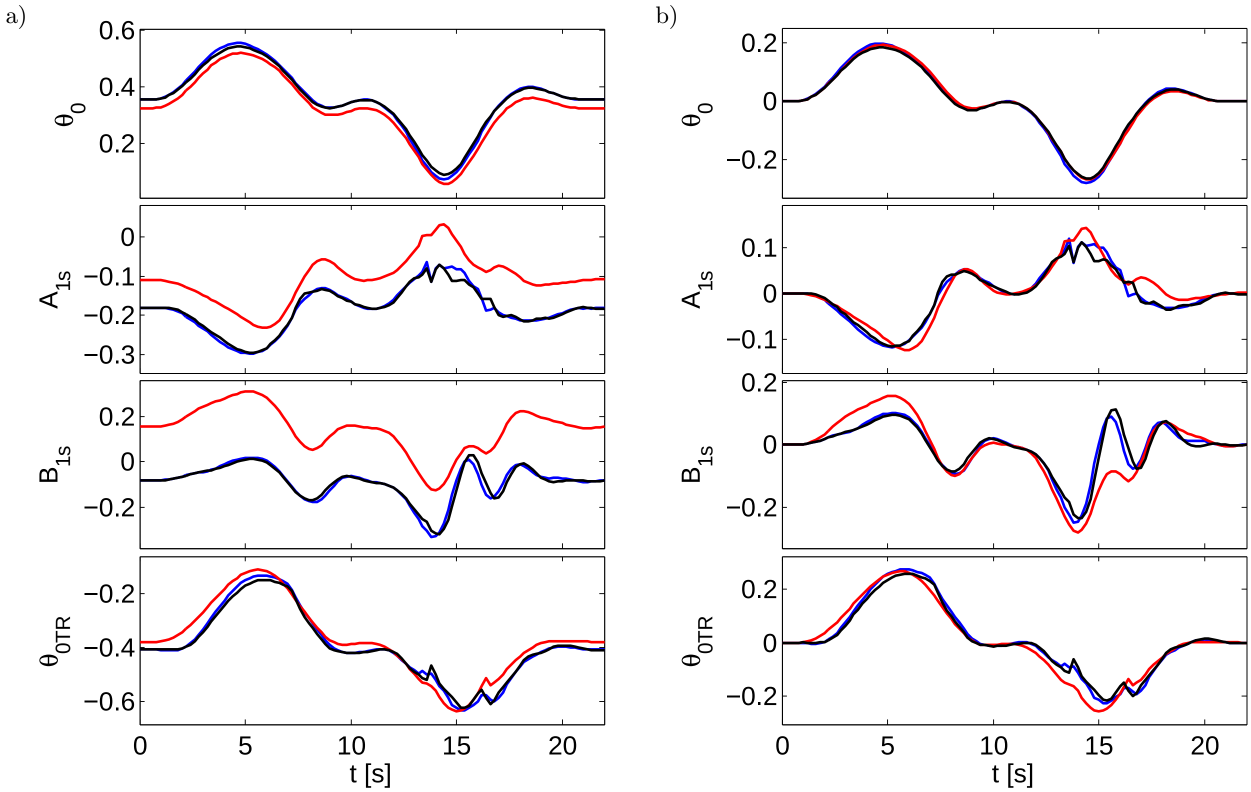

Figure 5. Command time-histories in hurdle-hop manoeuvre with minimum order inverse simulation model: (a) total command travel and (b) displacement from trim. – IS problem solved with reference model, – IS problem solved with lower order inverse model , – IS problem solved with MPC approach.

inverse simulation step. In this framework, the standard integration approach to the solution of the inverse simulation problem was modified in order to be more robust and flexible.

The approach was tested on a series of manoeuvres used for the analysis of rotorcraft handling qualities. The results show the ability of the MPC approach to generate trajectories and command time–histories very similar to the desired ones. Furthermore the approach was tested on a minimum complexity model and the performance of the technique proved to be still absolutely adequate.

Future work will be focused on the tuning of many parameters which are left to the user in the present techniques. Objective function, size of the receding horizon, states provided as feedback to the lower–order model can all be tuned to provide better performances. Further research will be also dedicated to the evaluation of the minimum order model capable of providing satisfactory results in the MPC approach, including linear ones.

References

[1]D.G. Thomson and R. Bradley, "Inverse simulation as a tool for flight dynamics research Principles and applications," Progress in Aerospace Sciences, Vol. 42, No. 3, May 2006, pp. 174-210.

[2]D.G. Thomson and R. Bradley, "Development and Verification of an Algorithm for Helicopter Inverse Simulation," Vertica, Vol. 14, No. 2, 1990, pp. 185-200.

[3]R.A. Hess, C. Gao, "A Generalized Algorithm for Inverse Simulation Applied to Helicopter Maneuvering Flight," Journal of the American Helicopter Society, Vol. 16, No. 5, 1993, pp. 3-15.

[4]Kato O. and Sugiura I, "An interpretation of airplane general motion and control as inverse problem," Journal of Guidance, Control& Dynamics, Vol. 9, No. 2, Mar.–Apr. 1986, pp. 198204.

[5]R.A. Hess, C. Gao, and S.H. Wang, "Generalised technique for inverse simulation applied to aircraft manoeuvres," Journal of Guidance, Control & Dynamics, Vol. 14, No. 5, Sep.–Oct. 1991, pp. 920-926.

[6]G. De Matteis, L.M. De Socio, and A. Leonessa, "Solution of aircraft inverse problems by local optimization," Journal of Guidance, Control & Dynamics, Vol. 18, No. 3, May–Jun. 1995, pp. 567-571.

[7]M. Borri, C.L. Bottasso, and F. Montelaghi, "Numerical Approach to Inverse Flight Dynamics," Journal of Guidance, Control & Dynamics, Vol. 20, No. 4, Jul.–Aug. 1997, pp. 742-747.

[8]S. Rutherford, and D.G. Thomson, "Helicopter Inverse Simulation Incorporating an Individual Blade Rotor Model," Journal of Aircraft, Vol. 34, No. 5, Sep.–Oct. 1997, pp. 627-634.

[9]G. Avanzini, and G. De Matteis, "Two-Timescale Inverse Simulation of a Helicopter Model," Journal of Guidance, Control & Dynamics, Vol. 24, No. 2, Mar.–Apr. 2001, pp. 330-339.

[10]M. Bagiev, D.G. Thomson, D. Anderson and D. Murray-Smith, "Hybrid inverse methods for helicopters in aggressive manoeuvring flight". In: 17th IFAC Symposium on Automatic Control in Aerospace, June 2007, Toulouse, France.

[11]G. Avanzini, G. De Matteis, and A. Torasso, "Comparison of Helicopter Trim Techniques," Proc. of the XX Conference of the Italian Association of Aeronautics and Astronautics (AIDAA), Milan, June 29 – July 3, 2009.

[12]G. Avanzini, G. De Matteis, and A. Torasso, "Modelling Issues in Helicopter Inverse Simulation," 36th European Rotorcraft Forum ERF2010, Paris, Sept. 7 – 9, 2010.

[13]J.J. Howlett, "UH-60A Black Hawk Engineering Simulation Program, Volume 1: Mathematical Model," NASA CR 166309, 1981.

[14]D.A. Peters, and N. HaQuang, "Dynamic inflow for practical applications," Journal of the American Helicopter Society, Vol. 33, No. 4, Oct. 1988, pp. 64-68.

[15]Anonymous, "Handling Qualities Requirements for Military Rotorcraft," US Army Aviation and Missile Command, ADS-33E-PRF, 2000.

[16]P.D. Talbot, B.E. Tinling, W.A. Decker, and R.T.N. Chen, "A mathematical model of a single main rotor helicopter for piloted simulation," NASA TM-84281, 1982.

[17]D.A. Peters, and D. Barwey, "A General Theory of Rotorcraft Trim", Mathematical Problems in Engineering, Vol. 2, 1996, pp. 1–34.

[18]J.S.G. McVicar, and R. Bradley, "Robust and efficient trimming algorithm for application to advanced mathematical models of rotorcraft", Journal of Aircraft, Vol. 32, No. 2, Feb. 1995, pp. 439–442.

[19]D.Q. Mayne, J.B. Rawlings, C.V. Rao, P.O.M. Scokaert, "Constrained model predictive control: Stability and optimality", Automatica, Vol. 36, N0. 6, June 2000, pp. 789–814.

[20]R.A. Hess, Y.C. Jung, "An Application of Generalized Predictive Control to Rotorcraft Terrain-Following Flight",IEEE transactions on systems, man, and cybernetics, Vol. 19, No. 5, Sept.–Oct 1989.

[21]Y.C. Jung, R.A. Hess, "Precise Flight-Path Control Using a Predictive Algorithm", Journal of Guidance, Control & Dynamics, Vol. 14, No. 5, Sep.–Oct. 1991, pp. 936-942.

[22]C.L. Bottasso, R. Nicastro, B. Savini, and L. Riviello, "Adaptive Reference-Augmented Predictive Control", Proceedings of the 2007 American Control Conference, New York City, USA, July 11-13, 2007