Predictive inverse simulation of helicopters in aggressive manoeuvring flight

M. Bagiev, D. G. Thomson and D. Anderson
Department of Aerospace Engineering
University of Glasgow
Glasgow, UK

D. Murray-Smith
Department of Electronics and Electrical Engineering
University of Glasgow
Glasgow, UK

ABSTRACT

A conventional inverse simulation does not accommodate control constraints; hence for aggressive manoeuvring flight conditions, where control inputs are close to the limits, these algorithms lose some of their applicability. A modification of the conventional inverse simulation technique that accommodates the onset of physical limits or constraints is proposed in this paper. In this way a process of constraints handling is incorporated into the inverse simulation algorithm. Therefore, the aim of this paper is to demonstrate that conventional inverse simulation can be improved in terms of the realism of the results by applying a predictive capability for applications involving manoeuvring flight. The paper gives details of the development of the predictive inverse simulation algorithm and helicopter model used and, by presenting examples of results calculated for pop-up and lateral realignment manoeuvres demonstrates that a 'receding horizon' predictive approach offers improvements in the realism of inverse simulation results.

NOMENCLATURE

H_p	prediction horizon (s)
h	height of pop-up manoeuvre (m)
h_1, h_2	desired and achieved altitude change in modified pop-up manoeuvre (m)
s	distance covered in a manoeuvre (m)
t	time (s)
t_{dp}	time at decision point (s)
t_m	total time to complete manoeuvre (s)
$\mathbf{u}$	control vector

Paper No. 3591. Manuscript received 13 May 2010, accepted 2 June 2011.

https://doi.org/10.1017/S0001924000006631 Published online by Cambridge University Press

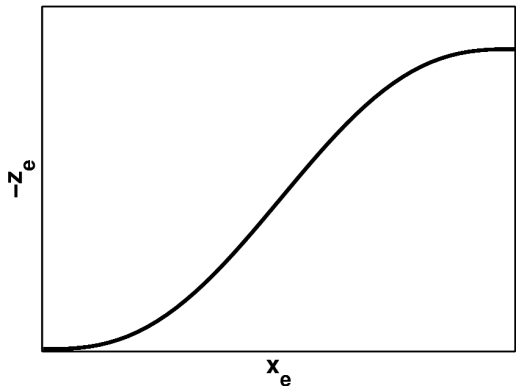

Figure 1. A profile for the pop-up manoeuvre.

$\mathbf{u}_{min}$,$\mathbf{u}_{max}$	control input limitations
V_f	flight velocity (kts)
V_{f1}, V_{f2}	initial and final flight velocity in a modified pop-up manoeuvre (kts)
$\mathbf{x}$	state vector
x_e, y_e, z_e	helicopter position relative to earth fixed frame of reference (m)
$\mathbf{y}$	output vector
$\mathbf{y}_d$	desired output vector
z_{dp},	position, velocity and acceleration referred to vertical earth z-axis direction at the decision point (m, ms^{-1}, m/s^2)
$\dot{z}_e$, $\ddot{z}_e$	vertical velocity and acceleration referred to earth fixed axes set (ms^{-1}, m/s^2)
τ	time elapsed since decision point (s)
τ_m	time from decision point to the end of the manoeuvre (s)
ψ	heading angle (deg)
ψ_1, ψ_2	original and desired heading angles in modified pop-up manoeuvre (deg)

1.0 INTRODUCTION

Helicopter inverse simulation allows calculation of the pilot control inputs that will force a helicopter to fly a specified manoeuvre. Helicopter inverse simulation was first used successfully by Thomson[1] to quantify helicopter agility, and since then it has been an active topic of research[2-8]. Perhaps the most comprehensive surveys of principles and applications of inverse simulation are those by Thomson and Bradley[9,10]. Although the various algorithms that have been developed have found a wide range of applications (for example, preliminary assessment of helicopter handling qualities and workload; conceptual design of helicopters[9,10]), they can produce unrealistic results in the form of control strategies at the edges of the flight envelope that would not normally have been adopted by an experienced pilot. This is particularly important in flight regimes where constraints, such as control limits, are approached. An inverse simulation is driven by a mathematical description of the manoeuvre of interest and if the defined manoeuvre is sufficiently aggressive, it is possible that the simulation may predict values that exceed the physical limits of the real vehicle. These limits might include mechanical limitations of the controls, the control rates (based on actuator saturation characteristics), limitations of rotor and tail rotor torque, human pilot limitations, and even structural limits of critical components. In this paper only the limitations of the helicopter mechanical controls are considered.

As an example, Fig. 1 shows a type of manoeuvre performed by a pilot in terrain following flight known as the pop-up manoeuvre.

https://doi.org/10.1017/S0001924000006631 Published online by Cambridge University Press

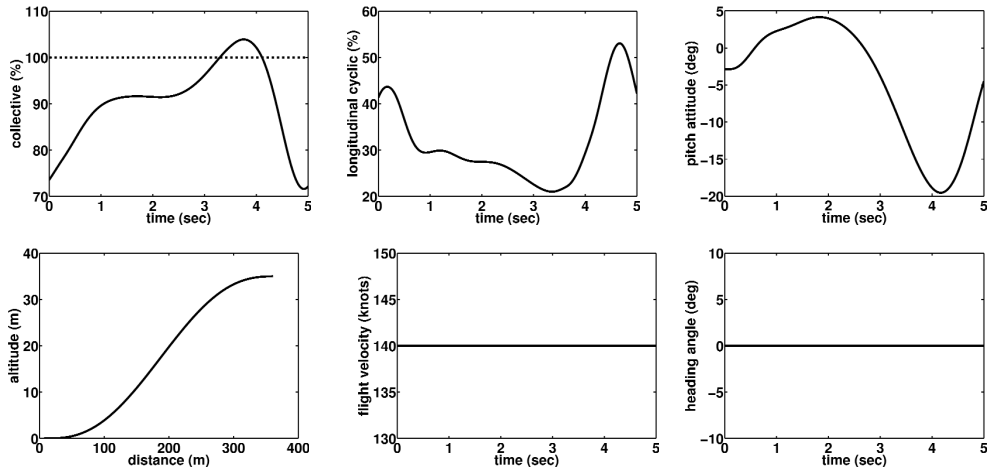

Figure 2. Inverse simulation results for a Lynx helicopter flying a pop-up manoeuvre (V_f = 140kt, h = 35m, t_m = 5 seconds).

Figure 2 shows the results of an inverse simulation of an AgustaWestland Lynx helicopter flying such a manoeuvre with a flight velocity V_f =140kts, an increase in altitude h = 35m, and a manoeuvre time t_m = 5 seconds. In this particular manoeuvre representation the velocity and heading are maintained at constant values throughout. Inspection of Fig. 2 illustrates that the main rotor collective pitch displacement exceeds its physical limit (dotted line) after 3·3 seconds, falling below the limit again after 4·2 seconds while all other controls remain within their limits. This may be an acceptable result for some circumstances (for studies focused on basic performance or flight dynamics, for example) but for applications requiring realistic strategies that mimic those of a real pilot it would clearly be unacceptable (handling qualities or control system design, for example). During an actual flight, at some point during the manoeuvre, the pilot would realise that the originally planned trajectory was unlikely to be achieved and would either adopt a less aggressive strategy or abort the task. It is also possible to limit a state value. In the example demonstrated in Fig. 2 the aircraft nose drops to –20 deg, which might be beyond an acceptable limit due, for example, to the limitations in the pilot view from the cockpit.

The pilot behaviour during such a manoeuvre is analogous to the 'receding horizon' concept that found a wide application in predictive control[11]. The pilot selects a control strategy based not only on the current flight state and his/her own internal model of the helicopter dynamic response, but on the likelihood of clearing the obstacle at some point in the future, or over a 'prediction horizon'. If some physical limit of the vehicle is exceeded over a prediction horizon, then a new control strategy must be applied. The integration-based inverse simulation algorithms[3,4] all operate using a similar approach, but with a much shorter prediction horizon than that of the pilot. However, as noted by Anderson[12], there is a strong similarity between the stability constraints on the algorithms in integration-based inverse simulation and those in nonlinear predictive control, suggesting possible synergies between these techniques.

The aim of this paper is to demonstrate that conventional inverse simulation can be improved in terms of the realism of the results by applying a predictive capability for applications involving helicopter aggressive manoeuvring flight. To achieve this, a rotorcraft inverse simulation package (RISP) was developed in the MATLAB environment, which includes a predictive inverse simulation algorithm (PRISM) with predictive capability (the receding horizon approach), a rotorcraft simulation model (RotorcraftSim), and a library of helicopter manoeuvres. In the following section the modification necessary to a standard inverse simulation algorithm to give it this predictive capability is discussed.

https://doi.org/10.1017/S0001924000006631 Published online by Cambridge University Press

2.0 PREDICTIVE INVERSE SIMULATION

2.1 Conventional inverse simulation

In general, aircraft dynamics may be described by the nonlinfear equations of motion in the following standard form of the initial value problem:

$$\dot{\mathbf{x}} = f(\mathbf{x},\mathbf{u}); \quad \mathbf{x}(0) = \mathbf{x}_0 \qquad \ldots (1)$$

$$\mathbf{y} = \mathbf{g}(\mathbf{x}) \qquad \ldots (2)$$

where $\mathbf{x}$ is the system state vector, $\mathbf{u}$ is the control vector, and $\mathbf{y}$ is the output vector. The aim of the inverse simulation algorithm is to calculate the control time histories $\mathbf{u}$ from a predefined, or desired, output vector $\mathbf{y}_d$. As demonstrated by Thomson and Bradley[9], Equations (1) and (2) may be recast by differentiating Equation (2):

$$\dot{\mathbf{y}} = \frac{d\mathbf{g}}{d\mathbf{x}}\dot{\mathbf{x}} = \frac{d\mathbf{g}}{d\mathbf{x}}\mathbf{f}(\mathbf{x},\mathbf{u}) \qquad \ldots (3)$$

If Equation (3) is invertible with respect to $\mathbf{u}$ it is possible to write:

$$\mathbf{u} = \mathbf{h}(\mathbf{x},\dot{\mathbf{y}}_d) \qquad \ldots (4)$$

and substituting in Equation (1) gives:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x},\mathbf{h}(\mathbf{x},\dot{\mathbf{y}}_d)) = \mathbf{F}(\mathbf{x},\dot{\mathbf{y}}_d) \qquad \ldots (5)$$

Equations (4) and (5) form a complete statement of the inverse problem with $\dot{\mathbf{y}}_d$ as the input vector and $\mathbf{u}$ as the output vector.

A conventional inverse simulation usually employs one of two different methods, numerical differentiation or numerical integration. The differentiation method was first used by Thomson[1] and the numerical integration technique for inverse simulation was proposed by Hess *et al*[13]. For the latter approach, the initial flight trajectory is divided into small time intervals and the nonlinear equations of motion are integrated and compared with desired trajectories over each interval. Typically, a Newton-Raphson iterative scheme is applied to minimise the error vector over each interval and thus over the trajectory as a whole. Rutherford and Thomson[4] used a similar approach in a numerical integration-based algorithm called GENISA (generic inverse simulation algorithm). A modified MATLAB version of this algorithm forms the basis of the new PRISM algorithm, which has the required predictive capability and this will be discussed in detail in the next subsection.

2.2 Modification of conventional inverse simulation to include predictive capability

A modification of the conventional inverse simulation technique that accommodates the onset of physical limits or constraints is proposed. A 'receding horizon' approach[11] is adopted in the algorithm to check if the constrained variables are within defined limits over a set period or prediction horizon (H_p) ahead of the current solution point. If they are, the PRISM algorithm moves to the next time point; if not, the remaining trajectory (recalling that this is used as an input for the inverse simulation algorithm) must be redefined with the aim of completing the task within the physical limits of the aircraft. From the inverse model defined by Equations (4) and (5), the task is to select $y_d(t)$ such that,

$$\mathbf{u}(t) \in [\mathbf{u}_{min}(t), \mathbf{u}_{max}(t)] \quad \forall t \qquad \ldots (6)$$

where $\mathbf{u}_{max}$ and $\mathbf{u}_{min}$ are the control input limitations.

https://doi.org/10.1017/S0001924000006631 Published online by Cambridge University Press

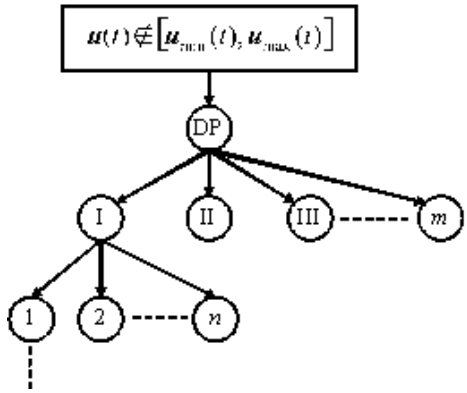

Figure 3. A diagram outlining the 'decision tree' approach adopted in the predictive inverse simulation (DP — decision point, I — flight velocity profile, II — heading/sideslip constraint, III — heading/sideslip profile).

Returning to the pop-up example, when the control limit is breached, there are a finite number of manoeuvre parameters, or modifiers, and their combinations may be adjusted to modify the remaining trajectory, for example the airspeed, geometry (height, longitudinal or lateral distance), or sideslip/heading constraint strategy. Often this will involve reducing the level of aggressiveness of the manoeuvre. The choice of sideslip or heading constraint has its origins in the inverse simulation algorithm, but does also reflect piloting strategy where a desired trajectory, or manoeuvre, can be flown controlling either heading or sideslip by using pedal inputs (i.e. tail rotor collective). By choosing one of these strategies, the calculated control inputs can be affected, especially in lateral manoeuvres, therefore making this a possible quantity to use in changes of 'piloting' strategy. Consider the case of a manoeuvre involving the helicopter in turning flight (a slalom, for example). Performing the manoeuvre with a constrained heading would imply that the nose of the helicopter is pointed in the same direction throughout the manoeuvre as the aircraft translates laterally – an unrealistic proposition. The more natural choice is for the sideslip constraint where, in effect, the centreline of the aircraft remains at a tangent to the flight path and the nose of the aircraft points along the track. The heading or sideslip angles can be constrained as a constant (usually zero), but also they can be defined as a profile, changing smoothly from the initial value to the defined one. For example, the pop-up manoeuvre can be flown with heading constrained at a fixed value (typically 0 deg), or the heading profile might be defined as some function such that it varies with time. This example will be demonstrated and discussed in the following section. Since the pop-up manoeuvre aims to avoid an obstacle or follow terrain, the possibility of altering the geometry of the manoeuvre to avoid breaching control limits is unlikely, unless some unexpected obstacle (for example, a tree at the top of the hill) is observed by the pilot after the manoeuvre has started.

The pop-up example presented in Fig. 2 demonstrated that the main rotor collective was exceeded. However, it is possible for other controls or constraints to be exceeded, and, as described previously, there are a number of manoeuvre parameters and combinations of these that could be modified. As there are several possible strategies that might lead to a successful manoeuvre, a 'decision tree' approach based on classical graph theory is ideal for this purpose. In many ways this approach mimics the actions of the pilot in that a 'decision point' is reached by projecting ahead into the future, and a suitable strategy can be chosen based on the search of the tree.

Approaches based on graph theory for aerospace control are well established, predominantly in the area of co-ordinated flight and multiple UAV trajectory/path planning[14]. One of the main challenges with this approach is the development of fast graph search algorithms and the selection of appropriate heuristics based on domain knowledge[15]. As the objective at this stage is not to compute the optimal change in trajectory but rather a feasible trajectory that would be more representative of those actually flown by pilots, a standard search algorithm[16] may be adopted. The flight profile modifiers are then carefully ordered based upon prior inverse simulation experience and known piloting strategies (the heuristics) to limit the average search time. In the examples evaluated in this study, the flight speed is always the first modifier to be checked, since the flight speed is the most likely parameter to influence proximity to control limits.

A diagram, presented in Fig. 3, demonstrates a general 'decision tree' approach implemented in the PRISM

https://doi.org/10.1017/S0001924000006631 Published online by Cambridge University Press

algorithm. As an example, only three quantities, or flight profile modifiers, were used for the first graph level (I is the flight velocity profile, II is the heading/sideslip constraint, and III is the heading/sideslip profile); while in general there might be m different modifiers. The second level of the graph includes all the possible variations of current modifier, including their combinations. The number of possible changes, n, can be reduced dramatically by having a priori knowledge, probably obtained from previous experience of similar situations. To achieve this, a database of inverse simulation results for different pop-up manoeuvres (i.e. different flight trajectories) was created and used as '*a priori* knowledge' to produce the examples presented in the paper.

Before examining results for the predictive inverse simulation it is essential to understand the nature of the rotorcraft mathematical model used, and how the flight trajectories are generated.

2.3 The Rotorcraft Simulation model

The RotorcraftSim model was developed in the MATLAB environment and forms a part of the RISP software. Use has been made of the helicopter mathematical model, HGS (Helicopter Generic Simulation)[17,18] . The HGS is a non-linear, seven degree of freedom, generic mathematical model, and was developed to be suitable for use in an inverse simulation. Multi-blade representations of the main and tail rotor were used, each blade being assumed rigid and to have constant chord and profile. The flow around the blades was assumed to be steady and incompressible, thus allowing two-dimensional aerodynamic theory to be applied in calculating the blade aerodynamic loads. Other significant features of the HGS include a dynamic inflow model, an engine model and look-up tables for fuselage, tailplane and fin aerodynamic forces and moments.

From a computational point of view the developed predictive inverse algorithm PRISM incorporating the RotorcraftSim with dynamic inflow model is very demanding. Therefore it was decided to use the Glauert model[19] to calculate the rotor induced velocity. The RotorcraftSim model has been verified against the original HGS model upon which it was based. Consistent results were observed across a wide range of test points giving confidence in the veracity of the new code. Given the very close match between results from RotorcraftSim and HGS, it was felt unnecessary to perform any new validation as the original model, HGS had been previously validated against flight test data[9]. The choice of this model was made on the basis that the focus of the research was on the development of a new approach to inverse simulation. As will become apparent, the method developed is not specific to the rotorcraft mathematical model and should be equally applicable to more comprehensive models which will have improved validity across a broader range of flight conditions.

2.4 Mathematical representation of helicopter manoeuvres

Thomson and Bradley[9,20] proposed and described, in detail, appropriate techniques for modelling helicopter manoeuvres and verified the validity of this approach by a comparison between flight test data and inverse simulation results. The same approach is adopted for the predictive inverse simulation.

It is assumed that the helicopter starts and finishes a defined manoeuvre under trimmed level flight conditions. Therefore, the vertical displacement relative to an earth fixed frame of reference, z_e, can be defined by a function of time, which has to satisfy six boundary conditions:

$$\begin{aligned} &\text{(i) } t = 0\text{: } z_e = 0, \quad \dot{z}_e = 0, \quad \ddot{z}_e = 0, \\ &\text{(ii) } t = t_m\text{: } z_e = -h, \quad \dot{z}_e = 0, \quad \ddot{z}_e = 0, \end{aligned} \qquad \ldots (7)$$

where t_m is the time taken to complete the pop-up manoeuvre, h is the height of the manoeuvre, and $\dot{z}_e$, $\ddot{z}_e$, are the vertical velocity and acceleration relative to an earth fixed frame of reference. The simplest function to satisfy these boundary conditions is a polynomial of order five. It was obtained in the form[9]:

$$z_e(t) = -\left[6\left(\frac{t}{t_m}\right)^5 - 15\left(\frac{t}{t_m}\right)^4 + 10\left(\frac{t}{t_m}\right)^3\right]h \qquad \ldots (8)$$

The vertical velocity, $\dot{z}_e$, and acceleration, $\ddot{z}_e$, can be obtained by differentiation of Equation (8). Longitudinal and lateral displacements of the helicopter can be determined from the expressions:

https://doi.org/10.1017/S0001924000006631 Published online by Cambridge University Press

$$x_e(t) = \int^t \sqrt{V_f(t)^2 - \dot{z}_e(t)^2}\,\mathrm{d}t \quad \ldots (9)$$

$$y_e(t) = 0 \quad \ldots (10)$$

However, by the nature of the proposed predictive inverse algorithm PRISM, a desired trajectory might be changed at some point during the manoeuvre, called the decision point, t_{dp} where the helicopter is in untrimmed flight. Therefore the remaining trajectory must be defined with a new set of six initial boundary conditions:

$$\begin{aligned} &\text{(i) } t = t_{dp}\text{:}\quad z_e = z_{edp},\quad \dot{z}_e = \dot{z}_{edp}\quad \ddot{z}_e = \ddot{z}_{edp};\\ &\text{(ii) } t = t_m\text{:}\quad z_e = -h,\quad \dot{z}_e = 0\quad \ddot{z}_e = 0 \end{aligned} \quad \ldots (11)$$

where z_{edp}, $\dot{z}_{edp}$, and $\ddot{z}_{edp}$ are the vertical displacement, velocity and acceleration at the decision point (relative to an earth fixed frame of reference). For simplicity in computation, the substitutions $\tau = t - t_{dp}$ and $\tau_m = t_m - t_{dp}$ are made, and boundary conditions (Equation (11)) are revised:

$$\begin{aligned} &\text{(i) } \tau = 0\text{:}\quad z_e = z_{edp},\quad \dot{z}_e = \dot{z}_{edp}\quad \ddot{z}_e = \ddot{z}_{edp};\\ &\text{(ii) } \tau = \tau_m\text{:}\quad z_e = -h,\quad \dot{z}_e = 0\quad \ddot{z}_e = 0 \end{aligned} \quad \ldots (12)$$

As in the previous example, the simplest function to satisfy these boundary conditions is a polynomial of order five:

$$z_e(\tau) = -\left(\frac{6(h+z_{edp})}{\tau_m^5} + \frac{3\dot{z}_{edp}}{\tau_m^4} + \frac{\ddot{z}_{edp}}{2\tau_m^3}\right)\tau^5 + \left(\frac{15(h+z_{edp})}{\tau_m^4} + \frac{8\dot{z}_{edp}}{\tau_m^3} + \frac{3\ddot{z}_{edp}}{2\tau_m^2}\right)\tau^4 -$$
$$\left(\frac{10(h+z_{edp})}{\tau_m^3} + \frac{6\dot{z}_{edp}}{\tau_m^2} + \frac{3\ddot{z}_{edp}}{2\tau_m}\right)\tau^3 + \frac{\ddot{z}_{edp}}{2}\tau^2 + \dot{z}_{edp}\tau + z_{edp} \quad \ldots (13)$$

The vertical velocity, $\dot{z}_e$, and acceleration, $\ddot{z}$, can be obtained by differentiation of Equation (13). Longitudinal and lateral displacements of the helicopter can be calculated from Equations (9) and (10).

3.0 SIMULATION RESULTS

3.1 Avoiding a collective pitch limit

Figure 4 shows the results of predictive inverse simulation applied to the example presented in Fig. 2 using a 2·5 seconds prediction horizon. In this case the impending collective limit breach is discovered at 0·8 seconds, which is 2·5 seconds before it occurs (dashed line), and a new strategy is adopted using the 'decision tree' approach implemented in the PRISM algorithm. The new strategy smoothly reduced the aircraft flight velocity by 2·5kts over a period of 4·2 seconds. Theoretically, decreasing the flight velocity would lead to a reduced collective angle being required to maintain the desired trajectory. Using a similar method as described for the pop-up manoeuvre in the previous section, a third-order polynomial was utilised to represent the change in flight velocity:

$$V_f(\tau) = -2\left(V_{f2} - V_{f1}\right)\left(\frac{\tau}{\tau_m}\right)^3 + 3\left(V_{f2} - V_{f1}\right)\left(\frac{\tau}{\tau_m}\right)^2 + V_{f1} \quad \ldots (14)$$

where τ and τ_m are as described previously, V_{f1} is the original flight velocity, and the V_{f2} is the desired flight velocity at the end of the manoeuvre.

In particular, in the example presented in Fig. 4, τ_m = 4·2 seconds, V_{f1} = 140kt, and V_{f2} = 137·5kt. Inspec-

https://doi.org/10.1017/S0001924000006631 Published online by Cambridge University Press

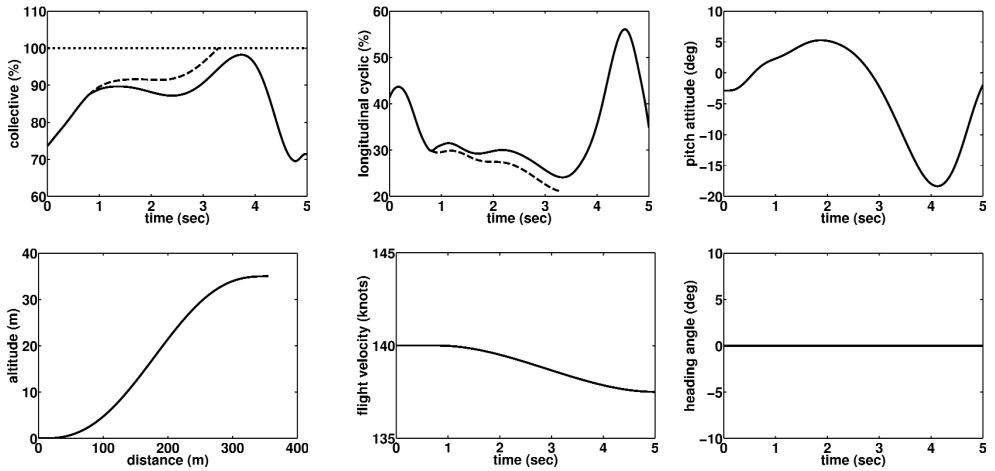

Figure 4. Predictive inverse simulation of a Lynx flying a pop-up manoeuvre (V_f = 140kt, h = 35m, t_m = 5 seconds) using a 2·5 seconds prediction horizon to accommodate control limits.

tion of Fig. 4 confirms that all of the controls, including the collective angle, remain within their limits. The heading angle is maintained at constant value, the desired height is achieved and, therefore, the manoeuvre is completed successfully.

3.2 Avoiding a longitudinal cyclic pitch limit

An inverse simulation result for another Lynx pop-up manoeuvre is shown in Fig. 5. In this example the task is to clear an obstacle at an altitude of 34m over a distance of 145m while maintaining a constant flight velocity of 60kt. As an example, the prediction horizon is chosen to be two seconds. It can be seen from Fig. 5 that a main rotor longitudinal cyclic limit would be exceeded after 4·2 seconds, but this was discovered two seconds before it occurs (dashed line).

In contrast to the previous example, a new trajectory was defined by changing a combination of two parameters presented in the 'decision tree' (Fig. 3), the flight velocity and heading angle profiles. The change in heading distributes the required pitching moment between longitudinal and lateral cyclic, decreasing the longitudinal cyclic and increasing the lateral cyclic. As the lateral velocity is zero in this manoeuvre, defining a change of 15 deg in heading is equivalent to imposing a sideslip angle of the same magnitude but in the opposite sense. To avoid increasing collective, the PRISM algorithm sets desired flight velocity to a lower value. It is worth reiterating at this point that the aim is to generate more realistic inverse simulation results and therefore the PRISM algorithm attempts to mimic the decision making process which would be taken by a pilot when attempting to modify his control strategy to complete a task.

As a result of the new algorithm, the helicopter flight speed is smoothly changed from 60 to 55kt and the heading angle from 0 to 15 deg. As in the previous example, a third order polynomial was utilised to represent the change in flight velocity (Equation 14) and heading:

$$\Psi(\tau) = -2\left(\Psi_2 - \Psi_1\right)\left(\frac{\tau}{\tau_m}\right)^3 + 3\left(\Psi_2 - \Psi_1\right)\left(\frac{\tau}{\tau_m}\right)^2 + \Psi_1 \quad 0 \le \tau \le \tau_m \qquad \ldots (15)$$

where τ and τ_m are as described previously, ψ_1 is the original heading, and ψ_2 is the desired heading at the end of the manoeuvre.

In this example, τ_m = 2·8 seconds, V_{f1} = 60 kts, V_{f2} = 55 kts, ψ_1 = 0 deg, and ψ_2 = 15 deg. It can be seen from Fig. 5, that such a combination of changes allows successful completion of the manoeuvre.

https://doi.org/10.1017/S0001924000006631 Published online by Cambridge University Press

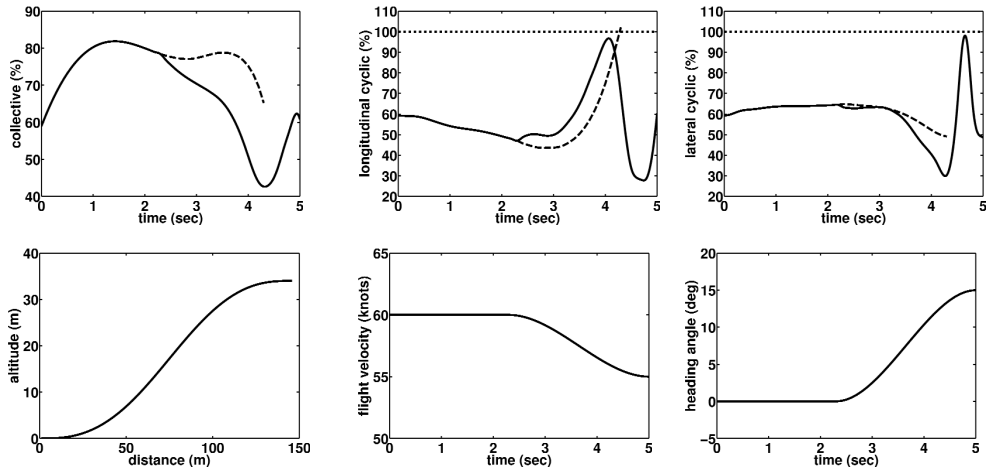

Figure 5. Predictive inverse simulation of a Lynx flying a pop-up manoeuvre (V_f = 60 kts, h = 34m, t_m = 5 seconds) using a two seconds prediction horizon to accommodate control limits.

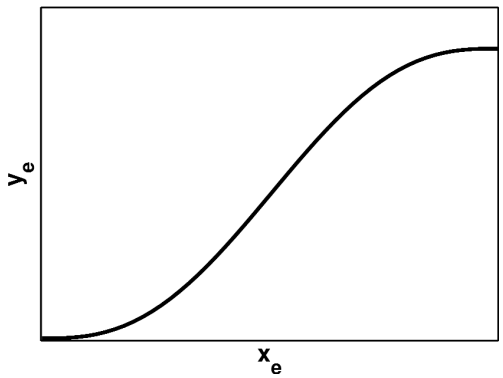

Figure 6. A profile for the lateral realignment manoeuvre.

3.3 Avoiding a lateral cyclic limit

A lateral realignment manoeuvre (Fig. 6) can be described as a longitudinal-lateral displacement of the aircraft at a constant altitude.

Figure 7 demonstrates the predictive inverse simulation results for the Lynx lateral realignment manoeuvre performed with airspeed, V_f = 100 kts, desired lateral distance, h_1 = 40m, and longitudinal distance, s =255m. The prediction horizon for this example was two seconds and the manoeuvre was performed with the sideslip constrained to be zero. A polynomial of order seven was used to represent the flight trajectory for this example. Figure 7 illustrates that the lateral cyclic pitch limit was exceeded at 2·2 seconds of flight time. The new, more realistic, piloting strategy chosen by the PRISM algorithm has changed the geometry of the manoeuvre, that is, reduced the desired lateral displacement from 40m (dashed line) to 30m (solid line). After applying the new strategy the maximum peak of the lateral cyclic was reduced to about 92% of the limit. The maximum values for the roll attitude angle were about ±43 degrees, and given the total time of five seconds, this represents a reasonably aggressive manoeuvre.

https://doi.org/10.1017/S0001924000006631 Published online by Cambridge University Press

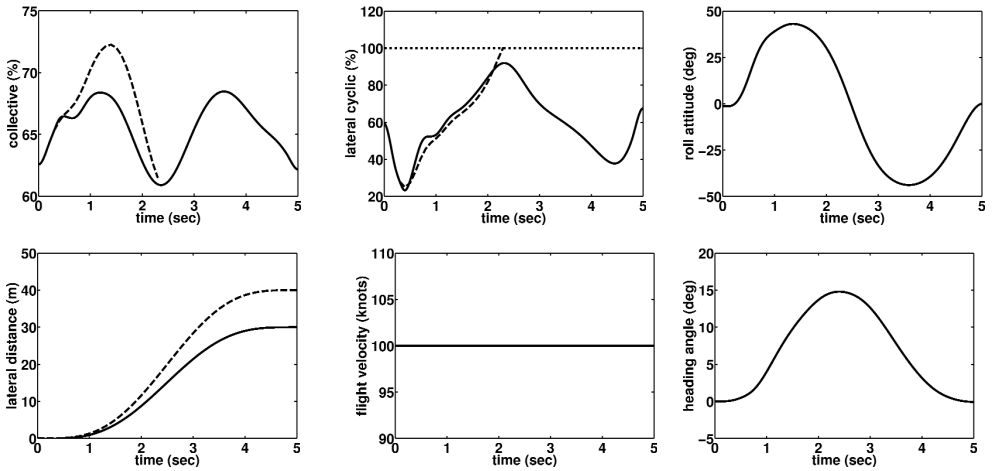

Figure 7. Predictive inverse simulation of a Lynx flying a lateral realignment manoeuvre (V_f =100kts, h_1 = 40m, h_2 = 30m, s = 255m).

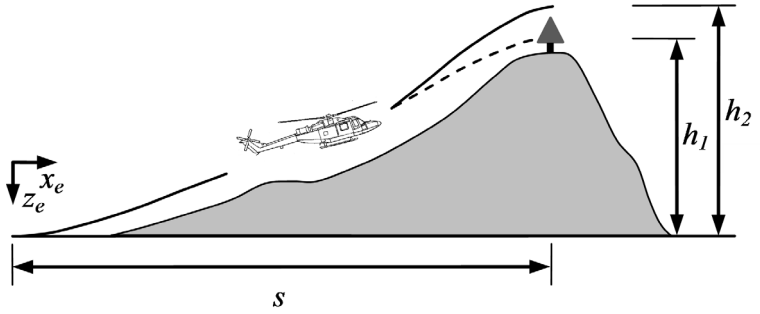

Figure 8. Pop-up manoeuvre with unexpected obstacle (a tree) at the top of the hill.

3.4 Effect of receding horizon length

The effect of receding horizon length is demonstrated for the scenario where a pilot is flying a helicopter over a hill or rise in the ground. At some point after starting a manoeuvre, the pilot observes an obstacle at the top of the hill (Fig. 8). The time interval between the decision point (i.e. point when the pilot observes the obstacle) and the point when this obstacle will be reached is defined as the 'visual prediction horizon' time. Therefore, to complete successfully the given task, the pilot must change the piloting strategy by increasing the desired height. Since the target height is increased the remaining trajectory will be more aggressive than the original one. To obtain realistic inverse simulation results, the receding horizon length can be adjusted to represent different pilot reaction time (related to piloting ability and workload) or visual cue environment (presence of fog, for example). Figure 8 depicts graphically this possible scenario of the pop-up manoeuvre, where the dashed line represents the original desired trajectory after the decision point, and the solid line represents a new flight trajectory.

The results of inverse simulation illustrate, as would be expected, that the sooner the pilot observes an unexpected obstacle (i.e. the longer the visual prediction horizon time) the higher the probability to complete successfully the given task. For example, Fig. 9 demonstrates results for a pop-up manoeuvre performed at 120kt with originally desired height, h_1 of 20m and a corrected desired height, h_2 of 30m (assuming that height of a tree is 10m) for two cases. In the first case (solid line) it is assumed that the pilot observed the tree 1 second into the manoeuvre time (i.e. the visual prediction horizon is four seconds), and in the second

https://doi.org/10.1017/S0001924000006631 Published online by Cambridge University Press

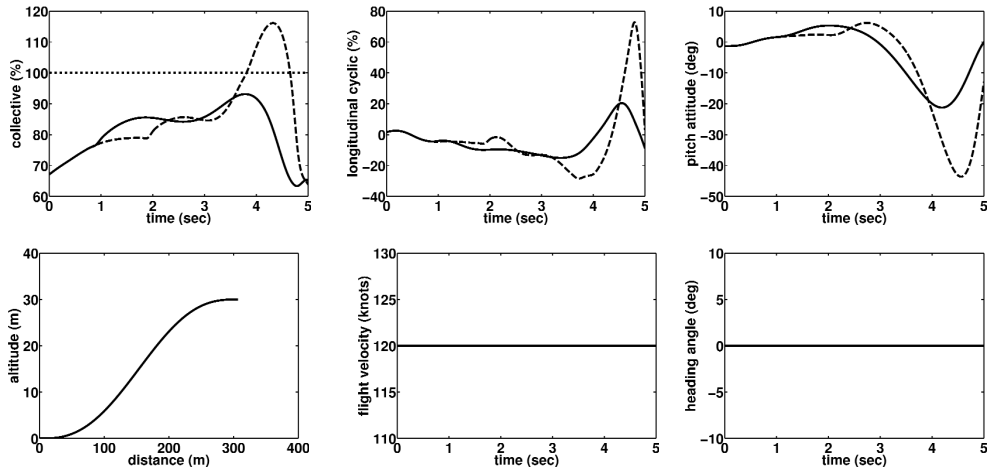

Figure 9. Predictive inverse simulation of a Lynx flying a pop-up manoeuvre (V_f = 120kt, h_1 = 20m, h_2 = 30m, t_m = five seconds) with unexpected obstacle (a tree with height of 10m) at the top of the hill.

case (dashed line) the pilot observed the tree two seconds into the manoeuvre time (i.e. the visual prediction horizon time is three seconds). It is clear from Fig. 9 that in the first scenario the task was successfully completed, while in the second scenario the pop-up manoeuvre cannot be completed since the main rotor collective limit is exceeded (also note large nose-down pitch displacement). In other words, if the pilot observed the tree one second into the manoeuvre time, it would be possible to change the desired height (or trajectory) and complete successfully the task. If the pilot observed the tree one second later, or two seconds into the manoeuvre time, the task would be aborted.

4.0 CONCLUDING REMARKS

It can be concluded that the predictive inverse simulation achieves the aim of improving the realism of inverse simulation results by implementing the 'receding horizon' approach. In this way a process of constraint handling is incorporated into the inverse simulation algorithm. One of the useful applications of conventional inverse simulation has been found to lie in the conceptual design of helicopters at the initial stages of the development of a new vehicle. The proposed predictive approach to inverse simulation should allow the number of iterative calculations during the design process to be reduced significantly by helping to identify the performance limitations of the proposed vehicle. More significantly however is the possible application of inverse simulation in areas such handling qualities assessment, pilot modelling, and advanced control system design as it allows more realistic pilot strategies to be determined. Further, in this work the constraints applied were on control input, and represent actual physical limits. Although the results in this paper demonstrate that alternative flight paths can be flown to achieve a particular end point without exceeding displacement limits, the ability of the control actuator to actually achieve these rapid motions is not considered. Including actuator dynamics in the model, and setting, for example, the control surface displacement rates would give even more realistic results, and a very effective way of designing the actuators necessary to achieve particular unsteady flight states.

The technique demonstrated in this paper extends the applicability of inverse simulation by broadening the range of manoeuvres that can be simulated. As well as an 'off-line' analysis tool, if implemented as a real-time simulation, this algorithm could be useful for on-board pilot aids, for example, as a trajectory generation algorithm for 'tunnel in the sky' guidance systems. The important feature which it has is that it allows vehicle limitations to be defined, and so any trajectory presented to the pilot (via the HUD, for example) should be feasible.

https://doi.org/10.1017/S0001924000006631 Published online by Cambridge University Press

ACKNOWLEDGMENTS

The authors gratefully acknowledge support from the UK Engineering and Physical Science Research Council through grant GR/S91024/01.

REFERENCES

1. THOMSON, D.G. An analytical method of quantifying helicopter agility, 1986, 12th European Rotorcraft Forum, Garmisch-Partenkirchen, Germany.
2. THOMSON, D.G. and BRADLEY, R. Development and verification of an algorithm for helicopter inverse simulation, *Vertica*, 1990, **14**, (2), pp 185-200.
3. HESS, R.A. and GAO, C. A generalized algorithm for inverse simulation applied to helicopter manoeuvring flight, *J American Helicopter Soc*, 1993, **16**, (5), pp 3-15.
4. RUTHERFORD, S. and THOMSON, D.G. Improved methodology for inverse simulation, *Aeronaut J*, 1996, **100**, (993), pp 79-86.
5. CAO, Y. A New Inverse solution technique for studying helicopter manoeuvring flight, *J American Helicopter Soc*, 2000, **45**, (1), pp 43-53.
6. CELI, R. Optimization-based inverse simulation of a helicopter slalom manoeuvre, *J Guidance, Control, and Dynamics*, 2000, **23**, (2), pp 289-297.
7. DOYLE, S.A. and THOMSON, D.G. Modification of a helicopter inverse simulation to include an enhanced rotor model, *J Aircr*, 2000, **37**, (3), pp 536-538.
8. AVANZINI, G. and DE MATTEIS, G. Two-timescale inverse simulation of a helicopter model, *J Guidance, Control, and Dynamics*, 2001, **24**, (2), pp 330-339.
9. THOMSON, D.G. and BRADLEY, R. The principles and practical application of helicopter inverse simulation, simulation practice and theory, *Intl J Federation of European Simulation Societies*, 1998, **6**, (1), pp 47-70.
10. THOMSON, D.G. and BRADLEY, R. Inverse simulation as a tool for flight dynamics pesearch — principles and applications, *Prog in Aerospace Sci*, 2006, **42**, (3), pp 174-210.
11. MACIEJOWSKI, J.M. *Predictive Control with Constraints*, 2001, Prentice Hall.
12. ANDERSON, D. Modification of a generalized inverse simulation technique for rotorcraft flight, Proceedings of the Institution of Mechanical Engineers, Part G: *J Aerospace Eng*, 2003, **217**, (2), pp 61-73.
13. HESS, R.A., GAO, C. and WANG, S.H., Generalized technique for inverse simulation applied to aircraft manoeuvres, *J Guidance, Control, and Dynamics*, 1991, **14**, (5), pp 920-926.
14. GRUNDEL, D., MURPHEY R. and PARDALOS, P.M. (Eds.) *Theory and Algorithms for Cooperative Systems*, 2004, Series on Computers and Operations Research, Vol. 4, World Scientific.
15. LUGER, G.F. and STUBBLEFIELD, W.A. *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*, 1999, Addison-Wesley.
16. SEDGEWICK, R. *Algorithms in C++, Part 5: Graph Algorithms*, 2002, Addison-Wesley.
17. THOMSON, D.G. Development of a generic helicopter model for application to inverse simulation, 1992, Internal Report No 9216, Department of Aerospace Engineering, University of Glasgow, UK.
18. PADFIELD, G.D. *Helicopter Flight Dynamics*, 1996, Blackwell Science.
19. GLAUERT, H. *A General Theory of the Autogyro*, 1926, Reports & Memoranda No 1111, Aeronautical Research Committee, London.
20. THOMSON, D.G. and BRADLEY, R. Mathematical definition of helicopter manoeuvres, *J American Helicopter Soc*, 1997, **42**, (4), pp 307-309.