JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 25, No. 6, November–December 2002

Engineering Notes

ENGINEERING NOTES are short manuscripts describing new developments or important results of a preliminary nature. These Notes cannot exceed 6 manuscript pages and 3 figures; a page of text may be substituted for a figure and vice versa. After informal review by the editors, they may be published within a few months of the date of receipt. Style requirements are the same as for regular contributions (see inside back cover).

Inverse Dynamics of Aircraft via Reduced-Order Shooting Methods

Franco Bernelli-Zazzera,* Gerlando Cappello,† and Paolo Mantegazza‡
Politecnico di Milano, 20158 Milan, Italy

I. Introduction

IN this Note, the inverse simulation of a military trainer aircraft is treated. The inverse simulation problem requires the determination of control inputs that will make a set of specified states follow a prescribed trajectory.

During the past decade, inverse simulation has received much attention. Applications include design of control inputs for aircraft maneuvers at large angles of attack, altitude programming in trajectory optimization, terrain following/avoidance problems, and many others. In recent years, various authors have presented different solution methods. Kato and Sugiura[1] proposed a differential approach in which the path constraints must be successively differentiated with respect to time until the control variables are included explicitly (see Ref. 2). With this approach, Kato and Sugiura[1] obtained slightly unrealistic results for some controls. Gao and Hess[3] proposed an integration algorithm that does not require time differentiation of the specified path constraints. The solution proposed by Sentoh and Bryson[4] is based on a simplification of the system dynamics, separating the force and the moment equations and expressing all quantities as functions of very few parameters. This approach cannot be generalized and introduces severe limitations because of the simplifications enforced on the mathematical model of the system.

More recently, two papers have appeared in which the inverse problem is discretized as a general optimization problem, with equality and inequality constraints that are functions of state variables. The proposed methods require neither any time differentiation of the trajectory constraints, nor the partial differentiation of output variables with respect to control inputs.[5,6] As reported in Ref. 7, the biggest issue that must be addressed when using an indirect method is the derivation of the necessary conditions themselves. For realistic trajectory simulations, the differential equations, path constraints, and boundary conditions can be complicated mathematical expressions or tabulated data to be interpolated. To impose the optimality conditions, it is necessary to differentiate these expressions analytically, and this limits the flexibility of the method. One of the limits of the earlier approaches[8] is the risk of obtaining solutions violating some physical constraints of the system, that is, large load factors or commands. The optimal control problem approach allows a better physical description of the problem, as well as the solution of multiobjective problems, for example, the minimization of the distance to a defined flight path with minimum fuel consumption. The other end of this approach is less straightforward and is more difficult than the approaches described before. One interesting use of numerical optimization is presented by Celi,[9] where the inverse problem is formulated using a particular set of path constraints that lead to families of acceptable solutions, among which the best one is selected on the basis of different performance criteria.

The method proposed here is even more general than the preceding because it allows an arbitrary combination of very general path and dynamics constraints, and, at the same time, it optimizes a defined performance index. The method departs from the classical approach for trajectory optimization based on direct transcription.[10] It arises from the concept of combining collocation and direct transcription and has the advantage of minimizing the number of variables and constraints involved in the numerical optimization with special regard to the equality constraints deriving from the direct transcription. This minimization of constraints and variables allowed the study of the inverse problem of the aircraft, including the dynamics of the actuators, on a personal computer, in a reasonable time, and without the need of using any specialized sparse optimization tool. It has been shown that the use of direct methods avoids both the explicit specification of, and the analytical conditions for, the Lagrange multipliers and the analytical differentiation of constraints on the path and on the state, thus making it simpler to use tabulated aeromechanics databases.

II. From Inverse Dynamics to the Optimal Control Problem

A. Optimal Control Problem

The optimal control problem can be formulated as follows: Given the dynamic system defined by a set of ordinary differential equations expressed in explicit form

$$\dot{\boldsymbol{y}} = \boldsymbol{f}[\boldsymbol{y}(t), \boldsymbol{u}(t), \boldsymbol{p}, t] \tag{1}$$

where $\boldsymbol{y}$ is the n_y-dimensional state vector, $\boldsymbol{u}$ an n_u-dimensional control vector, and $\boldsymbol{p}$ a parameter vector, find $\boldsymbol{p}$ to minimize the performance index

$$J = \phi(\boldsymbol{y}, t)|_{t_0}^{t_f} + \int_{t_0}^{t_f} L(\boldsymbol{y}, \boldsymbol{u}, \boldsymbol{p}, t)\,\mathrm{d}t \tag{2}$$

with constraints on the initial conditions at time t_0, $\psi_{0l} \le \psi[\boldsymbol{y}(t_0), \boldsymbol{u}(t_0), \boldsymbol{p}, t] \le \psi_{0u}$; the final conditions at time t_f, $\psi_{fl} \le \psi[\boldsymbol{y}(t_f), \boldsymbol{u}(t_f), \boldsymbol{p}, t] \le \psi_{fu}$; simple bounds on the state, $\boldsymbol{y}_l \le \boldsymbol{y}(t) \le \boldsymbol{y}_u$; simple bounds on the control, $\boldsymbol{u}_l \le \boldsymbol{u}(t) \le \boldsymbol{u}_u$; and a generic algebraic path constraint, $\boldsymbol{g}_l \le \boldsymbol{g}[\boldsymbol{y}(t), \boldsymbol{u}(t), \boldsymbol{p}, t] \le \boldsymbol{g}_u$, where ψ and $\boldsymbol{g}$ are vectors of dimensions n_ψ and n_g, respectively.

The basic idea is to minimize the distance between the real maneuver and an ideal one. This can be done by defining an ideal maneuver in terms of the vehicle trajectory and in terms of the state time history.

As reported in the literature,[6] when there is no idea of the reachability and controllability regions, it is better not to impose constraints on the states and instead describe the maneuver only through a performance index, that is, Eq. (2).

Received 23 March 2001; revision received 18 July 2002; accepted for publication 26 July 2002.

*Professor, Department of Aerospace Engineering, Via La Masa 34. Senior Member AIAA.

†Graduate Student, Department of Aerospace Engineering, Via La Masa 34.

‡Professor, Department of Aerospace Engineering, Via La Masa 34.

Table 1 NLP problem size

Method	N number of variables for the NLP	M number of constraints for the NLP	Size of J	Size of H
Complete direct transcription	3800	$2985 + M_{\text{ineq}}$	$3800 \times (2985 + M_{\text{ineq}})$	3800×3800
Alternative methods	800	M_{ineq}	$800 \times M_{\text{ineq}}$	800×800

B. Transcription Formulation

The basic approach for solving the optimal control problem by a direct transcription formulation will only be summarized because it has been presented in detail elsewhere.[10] All of the approaches divide the time interval into n_p segments of equal length, where the points are referred to as mesh or grid points. We introduce the notation $\boldsymbol{y}_j \equiv \boldsymbol{y}(t_j)$ to indicate the value of the state vector and $\boldsymbol{u}_j \equiv \boldsymbol{u}(t_j)$ to indicate the value of the control vector at a mesh point, so that the design variables vector becomes

$$\boldsymbol{x} = \lfloor \boldsymbol{y}_0, \boldsymbol{u}_0, \boldsymbol{y}_1, \boldsymbol{u}_1, \ldots, \boldsymbol{y}_{n_p}, \boldsymbol{u}_{n_p} \rfloor^T \tag{3}$$

The number of variables involved in the nonlinear programming (NLP) problem is $(n_y + n_u)(n_p + 1)$. The state equations are approximately satisfied by setting $\boldsymbol{y}_j = -\boldsymbol{y}_{j-1} - h_j(\boldsymbol{f}_j + \boldsymbol{f}_{j-1})/2$, for $j = 1, \ldots, n_p$. The step size is denoted by $h_j = t_j - t_{j-1}$, and the right-hand side of the differential equations (1) are given by $\boldsymbol{f}_j \equiv \boldsymbol{f}[\boldsymbol{y}(t_j), \boldsymbol{u}(t_j), t_j]$. Thus, the differential equations are replaced by the NLP equality constraints. The number of constraints deriving from the discretization is $n_y(n_p + 1)$, where n_y is the size of the state vector. In addition, there are the algebraic path constraints and the limitations on the controls and state variables, that is, Eqs. (5–7), discretized at each grid point.

C. Alternative Formulations

The aim of the alternative formulations is to minimize the number of variables and constraints involved in the NLP problem. In the first formulation, called optimized shooting, the state equations (1) are satisfied by numerical integration, separately from the NLP problem. In this way, both the number of equality constraints and the number of variables involved in the NLP problem decrease considerably. The equality constraints that are not satisfied are those on the final states and on the path. Only these equality constraints have to be satisfied during the NLP problem. Thus, only the control vector and the initial states are involved in the optimization process. The NLP variables are

$$\boldsymbol{x} = \lfloor \boldsymbol{y}_0, \boldsymbol{u}_0, \boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_{n_p} \rfloor \tag{4}$$

The states can be calculated by numerical integration. The step size used for the integration can be smaller than the discretization step size. This allows following the fast actuator's dynamics without increasing the number of variables involved in the NLP problem.

The second alternative formulation can be named implicit shooting. It arises from the consideration that all of the equality constraints deriving from the direct transcription constitute a nonlinear system of algebraic equations. The states, the controls, and the parameters of the dynamic system are the unknowns:

$$\begin{aligned}
&\psi_{0l} \le \psi[\boldsymbol{y}(t_0), \boldsymbol{u}(t_0), \boldsymbol{p}, t] \le \psi_{0u} \\
&\psi_{fl} \le \psi[\boldsymbol{y}(t_f), \boldsymbol{u}(t_f), \boldsymbol{p}, t] \le \psi_{fu} \\
&\boldsymbol{g}_l \le \boldsymbol{g}[\boldsymbol{y}(t), \boldsymbol{u}(t), \boldsymbol{p}, t] \le \boldsymbol{g}_u \\
&\boldsymbol{0} = \boldsymbol{y}_j - \boldsymbol{y}_{j-1} - \tfrac{1}{2}h_j(\boldsymbol{f}_j + \boldsymbol{f}_{j-1}) \qquad \Rightarrow \boldsymbol{G}(\boldsymbol{x}, \boldsymbol{z}) = \boldsymbol{0}
\end{aligned} \tag{5}$$

These unknowns must satisfy the nonlinear system introduced earlier. Thus, there will be as many dependent variables as there are equations derived from the equality constraints. The system of equations can be solved using a sparse solver, and so only the independent variables and the inequality constraints are involved in the NLP problem. The nonlinear system is solved with a Newton–Raphson technique.

This formulation allows for complete elimination of the equality constraints from the NLP problem and leaves the freedom to choose the independent variables involved in the optimization process. This is, indeed, a compromise between numerical stability and minimal sparsity loss.

If the matrix $\boldsymbol{A} = \partial \boldsymbol{G}/\partial \boldsymbol{z}$ is not singular, the variables chosen as independent are effectively independent, otherwise, another selection of variables must be performed. This matrix presents a very sparse structure because it is derived from the transcription of the differential equations.

The method allows for a considerable decrease in the number of equality constraints and variables involved in the NLP problem, thus allowing the use of a large number of robust, and widely available, general-purpose NLP solvers.

D. Optimization Process

The optimization was executed using a FORTRAN subroutine of the standard HARWELL™ mathematical library. This routine uses a sequential quadratic programming (SQP) approach to solve the NLP problem and does not take advantage of sparsity. The quantities that request considerable amounts of memory are essentially two: the Jacobian of the constraints and the Hessian of the augmented Lagrangian. These quantities take the form

$$[\boldsymbol{J}] = \frac{\partial c}{\partial \boldsymbol{x}} \text{ size} M \times N, \qquad [\boldsymbol{H}] = \frac{\partial^2 L^*}{\partial \boldsymbol{x}^2} \text{ size in the order of } N^2 \tag{6}$$

with

$$\boldsymbol{c} = \begin{bmatrix} \psi_{l_1} & \psi_{u_1} & \cdots & \psi_{ln_p} & \psi_{un_p} \end{bmatrix}^T \text{ size } M \times 1$$

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{u}_1 & \boldsymbol{u}_2 & \cdots & \boldsymbol{u}_{n_p} \end{bmatrix}^T \text{ size } N \times 1$$

where M is the total number of constraints, N is the total number of variables, and L^* is the augmented Lagrangian.

If a complete direct transcription formulation is used, these matrices are very sparse, and, for this type of application, it is not unusual to have just a small percentage of nonzero elements.[7] Consequently, exploiting sparsity to reduce both storage and computation time is a critical aspect of a successful implementation.

With the use of the alternative methods, the size of these matrices is considerably diminished, and the NLP problem can be solved without the use of sparse SQP. Table 1 gives an example of the dimensions of the preceding quantities for a maneuver of 10 s, with 20 grid points per second, 15 states, 4 controls, and an M_{ineq} number of inequality constraints. In practice, the alternative methods allow the use of a step size that can be greater than the collocation step size used for the complete direct transcription. In fact, the fast dynamics can be followed by integration in one case and with a finer collocation of the optimization process in the other case.

E. Jacobian of Constraints and the Gradient of the Objective Function

The computation of the gradient of the objective function and of the Jacobian of the constraints is a delicate task. There are two issues that must be considered. The first is related to the sensitivity of the variables. Early changes in the trajectory (near t_0) propagate to the end of the trajectory. The net effect is that the constraints can behave very nonlinearly with respect to the variables, thereby making the optimization problem difficult to solve. The second issue is the computational cost of evaluating the gradient information. Many approaches can be used to compute the gradient and Jacobian. In practice, for the case at hand, the most efficient method turned out to be the use of a finite difference technique. The central difference approximation to column J of the Jacobian matrix can be written

$$J_j = (1/2\|\delta_j\|)[c(x + \delta_j) - c(x - \delta_j)] \tag{7}$$

where J is the Jacobian matrix. To choose the finite difference perturbation size δ_j, the derivatives were calculated analytically by the formula

$$\frac{dy}{dt} = f[y(t), u(t), p, t], \qquad \frac{d}{du_k}\left(\frac{dy}{dt}\right) = \frac{d}{du_k}\{f[y(t), u(t), p, t]\}$$

$$\frac{d}{dt}\left(\frac{dy}{du_k}\right) = \frac{\partial f[y(t), u(t), t, p]}{\partial y}\frac{dy}{du_k} + \frac{\partial f[y(t), u(t), t, p]}{\partial u_k} \tag{8}$$

By numerical integration of the underlying differential equation, it is possible to calculate the derivative of the state y with respect to the command u_k. This process involves only one integration, but requires calculating analytically the derivative $\partial f/\partial y$ for each step of integration. Thus, it cannot be generalized.

The integrations for both analytical and central difference methods were done considering the physically realistic fact that, if a perturbation to the system occurs at time t_p, its effect on the system will propagate only from that instant until the final time. For this reason, it is not necessary to always integrate from the initial to the final time. Thus, if the perturbation occurs at time t_k, the integration must be carried out from t_{k-1} to t_f. This allows a considerable savings of computational time.

For the implicit shooting, the derivatives can be calculated by using the sparse solver again. In fact, it is necessary to calculate the variation of the dependent variables with respect to the independent variables. It is possible to write the following:

$$G[x, z(x)] = 0 \Rightarrow dG[x, z(x)] = 0$$

$$\frac{\partial G[x, z(x)]}{\partial z}\frac{\partial z}{\partial x_k} + \frac{\partial G[x, z(x)]}{\partial x_k} = 0$$

$$A\left(\frac{\partial z}{\partial x_k}\right) + \frac{\partial G[x, z(x)]}{\partial x_k} = 0 \tag{9}$$

is the resulting linear system.

The matrix A need not be calculated anew because it remains the same as that of the last iteration, which was used when solving the nonlinear system with the Newton–Raphson technique. In this way, the linear system can be solved with a sparse solver, changing and calculating only the partial derivative $\partial G/\partial x_k$. The process must be repeated for all of the independent variables.

III. Numerical Results

In this work, a number of maneuvers were studied. Some of them can be considered aerobatics, for example, looping and multiple roll, and some conventional, for example, flight level change or overcoming an obstacle. Only the results relative to the multiple roll will be presented here.

The type of description adopted is highly correlated to the method used for studying the inverse problem. In the numerical optimization, there are two elements that describe the entire maneuver: equality and inequality constraints and the objective function. The guiding line followed in this work was to constrain as few states as possible. In fact, constraining too many states on a realistic model of the aircraft is the most probable source of difficulties in solving the problem. Moreover, the conventional controls of the aircraft under consideration make it theoretically impossible to assign independently all six rigid-body degrees of freedom, furthermore forcing the need to relax the number of constrained states. Thus, the equality constraints are imposed only on a reduced set of transcription points, their number and placement chosen based on an educated guess to allow only minor violations of the equality constraints at all of the other points. Conversely, inequality constraints are retained at all of the grid points. Varying the lower and upper bounds of the constrained quantities allows a good description of the maneuver without forcing the quantities strictly to take a particular value, thus allowing an alternate approach to a reduced set of equality constraints. The choice of the objective function is not unique and can highly influence the capability of finding a solution. To take into account the physical limits of the aircraft under consideration, inequality constraints have been considered on the load factors, and constraints on the controls were included in terms of the real maximum deflections of aileron, rudder, elevator, and maximum thrust. These constraints guaranteed the physical implementation of the solution. Limitations on the maximum rates of controls were not included because the dynamics of the actuators was taken into account, and stall of the actuator was considered avoided by an adequate design:

$$-30 < \delta_a < +30 \text{ deg}, \qquad -30 < \delta_r < +30 \text{ deg}$$

$$-10 < \delta_e < +30 \text{ deg}, \qquad 0 < \delta_T < 2, \qquad -4 < n_z < +8 \tag{10}$$

The ideal maneuver consists of a continuous roll of the aircraft, whereas its center of gravity sustains a straight path at a constant altitude. This ideal trajectory can be followed exactly only by using unconventional controls, that is, by producing six independent commands. The literature reports many analytical definitions of the maneuver,[8,11] and the methods proposed for its inverse solution often constrain quantities that should be unconstrained in a conventional aircraft. These constraints can produce two effects. The first is a nonfeasible solution, that is, commands larger than physically allowable. The second effect is represented by large deviations from the rectilinear path. For these reasons, instead of constraining some specific quantity, the maneuver has been represented via a suitable cost function with the addition of inequality constraints. The objective function for this maneuver weighs suitably both the shifting of the real roll angle and the shifting of the real position from the ideal ones. In addition, inequality constraints were added to limit the maximum shifting from the ideal straight path. The ideal maneuver has then been described as follows:

$$\begin{cases} \phi_{\text{ideal}} = 0 & \text{for} \quad t < t_1 \\ \phi_{\text{ideal}} = \omega_{\text{ideal}}(t - t_1) & \text{for} \quad t \geq t_1 \end{cases}$$

$$\mathbf{PATH}_{\text{ideal}}(t) = \begin{cases} x_{\text{ideal}}(t) = \bar{V}t & \text{with} \quad \bar{V} \text{ const} \\ y_{\text{ideal}}(t) = 0 & \forall t \\ z_{\text{ideal}}(t) = \bar{z}_{\text{ideal}} & \text{const} \end{cases} \tag{11}$$

The objective function has been expressed as

$$J = \int_{t_0}^{t_f} \left\{ a[\mathbf{PATH}(t) - \mathbf{PATH}_{\text{ideal}}(t)]^T [\mathbf{PATH}(t) - \mathbf{PATH}_{\text{ideal}}(t)] + b[\phi(t) - \phi_{\text{ideal}}(t)]^2 \right\} dt \tag{12}$$

The adjustment of the weights a and b can determine whether it is more important to follow a rectilinear path at constant altitude or to follow the roll angle. For almost all of the simulations, the weight b on the roll angle was larger than the weight on the path. In fact, the shifting from the ideal path was controlled via an inequality constraint. The maximum allowed shift has been set to 4 m in each direction, with no further constraint imposed.

The roll rate was fixed at a constant value equal to $\pi/2$ rad/s. The initial states have been fixed equal to the states of the trim condition at the simulation altitude and velocity. For this reason, from the initial time t_0 to the intermediate time t_1, the ideal roll angle had been fixed to zero. When this was done, the initial states did not enter into the optimization variables, and it was not necessary to calculate the controls sequence to pass from a trimmed condition to the initial condition determined by the optimization routine. The command collocation time step is 0.25 s, whereas the integration time step is 11 times smaller. This allows a precise integration of the system dynamics, including actuator transients, with no extra optimization burden.

The results are presented in Figs. 1–3. In particular, Fig. 1 reports the computed optimal controls and the corresponding roll parameters, whereas Fig. 2 shows the most relevant trajectory parameters. The other states are not reported for the sake of conciseness

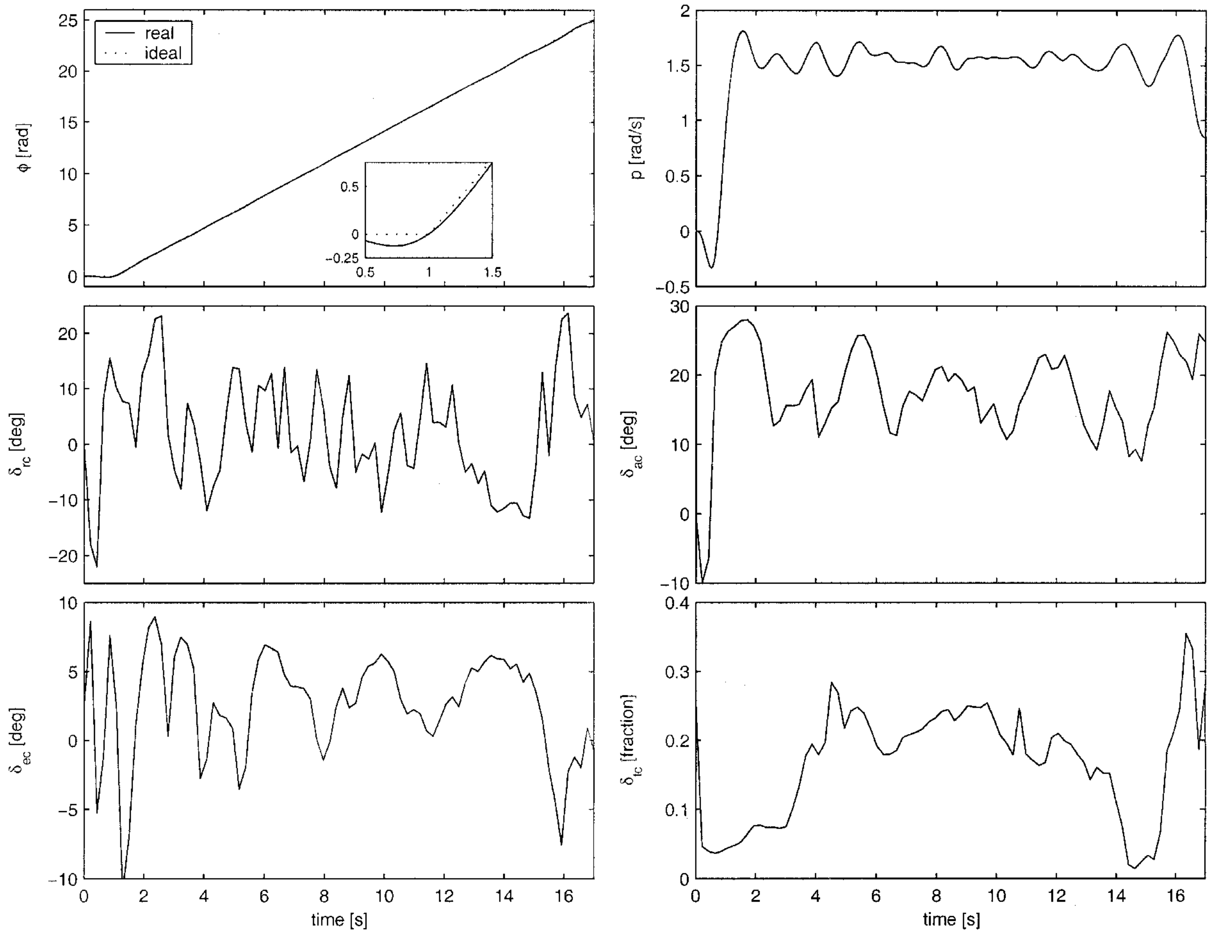

Fig. 1 Controls and relevant parameters of multiple-roll maneuver.

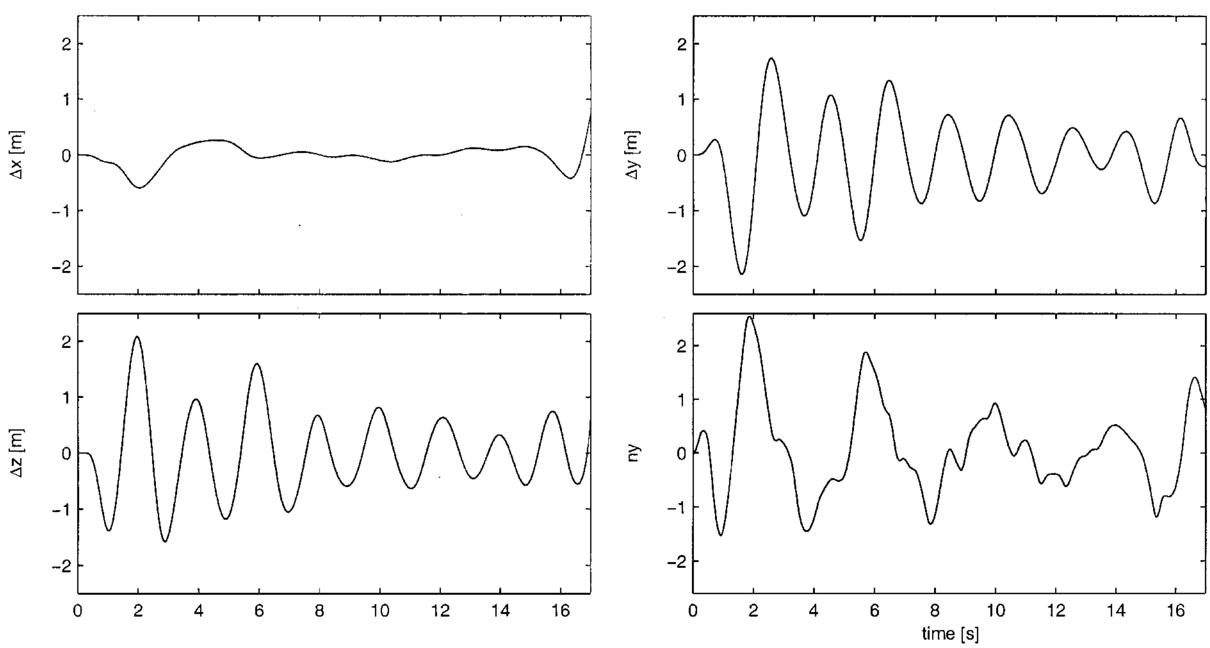

Fig. 2 Trajectory parameters of multiple-roll maneuver.

because they add no significant contribution to the discussion of the results. Figure 3 shows the attitude and flight path at a few selected points, namely, just before the start of the nominal roll maneuver (time = 0.97 s), at the end of the second turn (time = 8.99 s), at half of the last turn (time = 14.9 s), and just before the end of the maneuver (time = 16.7 s).

The analysis of the actual roll angle and followed path clearly show that the computed solution is very close to the ideal one. The greater weight assigned to the roll angle in the cost function makes it follow the ideal time history very well. Nevertheless, the displacement from the rectilinear flight path is fully acceptable. All of the commands are obviously within the imposed limits. Two considerations deserve special discussion. The first deals with the quasi-periodic nature of the maneuver, which does not, however, produce periodic control commands. A fundamental frequency appears in the aileron and elevator controls, but it is not evident in

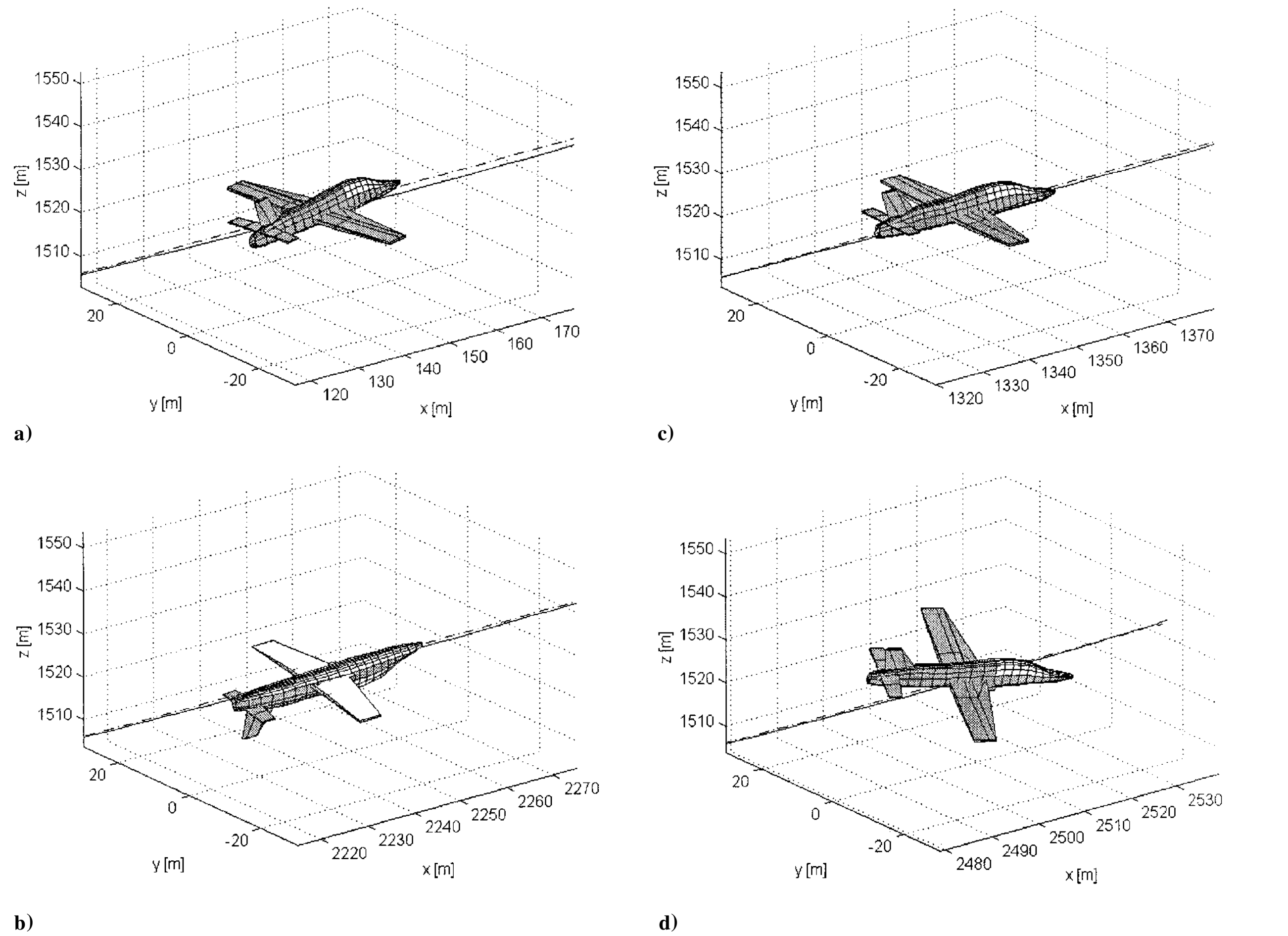

Fig. 3 Selected frames of multiple roll: a) time = 0.97 s, actual roll angle = −0.994 deg, desired roll angle = 0 deg; b) time = 14.9 s, actual roll angle = 1257.1 deg, desired roll angle = 1256.8 deg; c) time = 8.99 s, actual roll angle = 719.42 deg, desired roll angle = 719.89 deg; and d) time = 16.7 s, actual roll angle = 1416.4 deg, desired roll angle = 1415.3 deg.

the thrust and rudder commands. This is because there is no periodic constraint on the controls, and the influence of the initial and final transients propagates throughout the entire maneuver. A second interesting consideration is related to the initial and final transients. The conventional aircraft configuration at hand, as well as the limited control authority, make it impossible to produce abrupt variations in the roll rate alone. Thus, to maximize the overall performance, the aircraft has to "anticipate" the maneuver and end it with a considerable residual sideslip and angle of attack. This also produces the initial and final peaks in the lateral load factor.

IV. Conclusions

The algorithms presented as an alternative to a classical direct transcription helped to obtain good solutions to the inverse control of conventional aircraft for the execution of unconventional maneuvers. The proposed method is sufficiently robust to face the lack of an appropriate initial guess solution, and the computational expense is affordable.

The methods proposed rely on the physical fact that the real system behavior will be influenced by a limited number of parameters, far lower than the total number of unknowns of a complete transcription formulation. This is coherent with the evolution of a dynamic system being influenced by the boundary conditions, especially the initial ones, and the control inputs.

The methods offer the great advantage of eliminating the latter from the optimization process, even if the solution of the nonlinear system resulting from the assembled constraints is not trivial.

A further improvement could be represented by implementing an adaptive scheme to determine the state and control collocation grid.

References

[1] Kato, O., and Sugiura, I., "Introduction of Airplane General Motion and Control as Inverse Problem," *Journal of Guidance, Control, and Dynamics*, Vol. 9, No. 2, 1986, pp. 198–204.

[2] Bryson, A. E., and Ho, Y.-C., *Applied Optimal Control: Optimization, Estimation, and Control*, Wiley, New York, 1975.

[3] Gao, C., and Hess, R. A., "Inverse Simulation of Large-Amplitude Aircraft Maneuvers," *Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 4, 1993, pp. 733–737.

[4] Sentoh, E., and Bryson, A. E., "Inverse and Optimal Control for Desired Outputs," *Journal of Guidance, Control, and Dynamics*, Vol. 15, No. 3, 1992, pp. 687–691.

[5] Lee, S., and Kim, Y., "Time-Domain Finite Element Method for Inverse Problem of Aircraft Maneuvers," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 1, 1997, pp. 97–103.

[6] Lichtsinder, A., Kreindler, E., and Gal-Or, B., "Minimum-Time Maneuvers of Thrust-Vectored Aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 2, 1998, pp. 244–250.

[7] Betts, J. T., "Survey of Numerical Methods for Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 2, 1998, pp. 193–207.

[8] Hess, R. A., Gao, C., and Wang, S. H., "Generalized Technique for Inverse Simulation Applied to Aircraft Maneuvers," *Journal of Guidance, Control, and Dynamics*, Vol. 14, No. 5, 1991, pp. 920–926.

[9] Celi, R., "Optimization-Based Inverse Simulation of a Helicopter Slalom Maneuver," *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 2, 2000, pp. 289–297.

[10] Betts, J. T., and Kramer, E. J., "Application of Direct Transcription to Commercial Aircraft Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 18, No. 1, 1995, pp. 151–159.

[11] De Matteis, G., De Socio, L. M., and Leonessa, A., "Solution of Aircraft Inverse Problem by Local Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 18, No. 3, 1995, pp. 567–571.