JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 20, No. 1, January–February 1997

Time-Domain Finite Element Method for Inverse Problem of Aircraft Maneuvers

Suchang Lee* and Youdan Kim†
Seoul National University, Seoul 151-742, Republic of Korea

A method for solving nonlinear inverse problems is proposed. The inverse problem is formulated as a general optimization problem with equality constraints that are functions of state variables. The optimality conditions are derived by a variational approach. A time-domain finite element method is used to discretize the derived governing equations. The proposed method can utilize the control redundancy in a redundant case through the proper selection of the performance index and constraints. Time differentiation of trajectory constraints and partial differentiation of output variables with respect to the control inputs are not required, and there is no numerical integration in the developed algorithm. Examples of inverse solutions for an aileron roll maneuver and a bank-to-bank maneuver of aircraft are presented.

Nomenclature

$d\boldsymbol{x}(t_f)$, $\delta\boldsymbol{x}(t_f)$	= final state variation when holding final time fixed, and when final time is allowed to vary, respectively
$\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t)$	= system dynamics vector function
J	= performance index
$L(\boldsymbol{x}, \boldsymbol{u}, t)$	= Lagrangian
$\boldsymbol{S}(\boldsymbol{x}, t)$	= state constraint vector function
t	= time
$\boldsymbol{x}(t)$	= state vector
$\boldsymbol{u}(t)$	= control input vector
$\boldsymbol{\alpha}, \boldsymbol{\beta}(t), \boldsymbol{\nu}, \boldsymbol{\lambda}(t)$	= Lagrange multipliers
τ	= nondimensional elemental time
$\phi(\boldsymbol{x}, t)$	= final state weighting function
$\psi(\boldsymbol{x}, t)$	= final stage constraint

Superscript

T	= transpose operator

I. Introduction

THE inverse simulation problem involves determining the required control inputs that will make the specified states follow the prescribed state trajectories. During the last decade, the inverse simulation problem of aircraft maneuver has received much attention[1–6]: design of control inputs for aircraft nonlinear large-angle maneuvers, altitude programming in trajectory optimization,[1] terrain following/avoidance problem,[2] and others.

Kato and Sugiura[3] proposed a differential approach in which the path constraint must be successively differentiated with respect to time until the control variables are included explicitly. They applied it to the aileron roll maneuver of an A4D aircraft, however; they obtained slightly unrealistic results: rudder angles in excess of 30 deg, sideslip angles of almost 20 deg. Sentoh and Bryson[4] formulated the same problem using an optimal control algorithm that minimizes the integral of weighted square sum of deviation from a straight flight path and control surface deflections. To get a feasible solution, large weightings are imposed on control deflections and, therefore, the bank angle did not follow the specified trajectory. Gao and Hess[5] proposed an integration algorithm, which does not require time differentiation of the specified path constraints. Both the nominal case in which the number of control inputs equals the number of path constraints, and the redundant case, where the number of control inputs is greater than the number of path constraints, can be treated through the pseudoinverse concept. However, the partial derivatives of output variables with respect to the control inputs have to be evaluated numerically. No specific method was presented to exploit the control redundancy, and low-pass filtering techniques should be used to remove the high-frequency oscillations of the obtained control inputs. Matteis et al.[6] presented another integration approach using the local optimization concept in which the control redundancy could be exploited by adding new path constraints. However, the second partial derivatives of the output variables with respect to each control input must be evaluated numerically, and construction of the new path constraints for a specific performance requirement is complicated. The latter is a severe problem, because the path constraints may restrict the feasible region and sometimes a solution cannot be found through optimization.

In this paper, the inverse problem is formulated as a general optimization problem with equality constraints that are functions of state variables. The control redundancy can be fully utilized through the proper selection of the performance specifications and constraint equations. The proposed method does not require the time differentiation of trajectory constraints nor the partial differentiation of output variables with respect to the control inputs. Numerical differentiation is not involved in the procedure and, therefore, the ill-conditioning and initial guess sensitivity problems of numerical procedures can be avoided.

Hodges and Bless[7] derived the optimality conditions of the basic optimal control problem using a variational approach based on the Hamilton's weak principle and solved the governing equations by a time-domain finite element method. This methodology was applied to simulate nonlinear dynamic systems and to solve optimal control problems such as the trajectory optimization problem of a point-mass launch vehicle. This technique was extended to cope with the inequality constraints on state and control variables[8] and discontinuities in system dynamics.[9,10] Also, it was used to solve the trajectory optimization of a multistage launch vehicle with dynamic pressure constraints.[11–13]

A new method, expanding the Hodges and Bless method, is proposed to cope with the equality constraints on function of state variables, for solving nonlinear inverse problems. In the following section, the inverse problem is formulated as a general optimization problem. The optimality conditions are derived using a variational approach. In Sec. III, a set of nonlinear algebraic equations are derived through the discretization of the governing equations by the time-domain finite element method. Also, a numerical procedure to solve the discretized governing equations is discussed in detail. In

Received March 1, 1996; presented as Paper 96-3701 at the AIAA Guidance, Navigation, and Control Conference, San Diego, CA, July 29–31, 1996; revision received Aug. 5, 1996; accepted for publication Oct. 1, 1996.

*Graduate Student, Department of Aerospace Engineering. Member AIAA.
†Associate Professor, Department of Aerospace Engineering. Member AIAA.

Sec. IV, numerical examples of an aileron roll and a bank-to-bank maneuver of an A4D aircraft are presented. In Sec. V, the features of the proposed method are summarized.

II. Governing Equations for Inverse Problem

The inverse problem is one in which the system and the desired state trajectories are given and the control inputs have to be determined. We consider the nonlinear time-varying dynamic systems of the form

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \tag{1}$$

where $\boldsymbol{x} \in R^{n\times 1}$ denotes the state vector and $\boldsymbol{u} \in R^{m\times 1}$ denotes the control vector. Assume that the specified trajectories are represented as follows:

$$\boldsymbol{S}(\boldsymbol{x}, t) = 0, \qquad t \in [t_0, t_f] \tag{2}$$

Note in this equation that the control vector does not appear explicitly as in many realistic problems. Only the cases that the number of controls (m) equals or exceeds the number of constraint equations (l) will be treated. Initial conditions of the system and physical limitations on actuators should be carefully considered to determine the proper form of the desired trajectories, or unrealistic solutions may be obtained. Therefore, it is assumed that the specified constraints of Eq. (2) are consistent with the initial conditions of the system of Eq. (1).

We propose a method to solve the inverse problem by utilizing algorithms for optimization. Consider a general constrained optimization problem with the following performance index, system dynamics of Eq. (1), and final time constraints of Eq. (4). The imposed final time constraints have to be consistent with the specified state trajectories with the consideration of the time derivatives of related state variables. At the same time, it must properly describe the desired final condition of the dynamic system

$$J_0 = \int_{t_0}^{t_f} L(\boldsymbol{x}, \boldsymbol{u}, t)\, \mathrm{d}t + \phi(\boldsymbol{x}, t)\big|_{t_0}^{t_f} \tag{3}$$

$$\psi[\boldsymbol{x}(t_f), t_f] = 0 \tag{4}$$

The inverse problem can be formulated as an unconstrained minimization problem via direct augmentation method.[14,15] The constraint equations are augmented as follows:

$$J' = \int_{t_0}^{t_f} [L + \boldsymbol{\lambda}^T(\boldsymbol{f} - \dot{\boldsymbol{x}}) + \boldsymbol{\beta}^T \boldsymbol{S}]\, \mathrm{d}t + \Phi\big|_{t_0}^{t_f} \tag{5}$$

$$\Phi = \phi(\boldsymbol{x}, t) + \boldsymbol{\nu}^T \psi(\boldsymbol{x}, t) \tag{6}$$

where $\boldsymbol{\lambda}(t) \in R^{n\times 1}$, $\boldsymbol{\beta}(t) \in R^{l\times 1}$, and $\boldsymbol{\nu} \in R^{q\times 1}$ are Lagrange multipliers. Note that a similar problem can be formulated by minimizing weighted least square error of the path constraints as in the model following or the model reference control technology. However, exact following of the path constraints is not guaranteed, and initial transient error is unavoidable.

The states should be continuous at the initial and final times. Implying weak boundary conditions on these variables,[7] the new performance index takes the form

$$J = \int_{t_0}^{t_f} [L + \boldsymbol{\lambda}^T(\boldsymbol{f} - \dot{\boldsymbol{x}}) + \boldsymbol{\beta}^T \boldsymbol{S}]\, \mathrm{d}t + \Phi\big|_{t_0}^{t_f} + \boldsymbol{\alpha}^T(\boldsymbol{x} - \hat{\boldsymbol{x}})\big|_{t_0}^{t_f} \tag{7}$$

where

$$\begin{aligned} \boldsymbol{x}|_{t_0} &\equiv \lim_{t\to t_0^+} \boldsymbol{x}(t), & \boldsymbol{x}|_{t_f} &\equiv \lim_{t\to t_f^-} \boldsymbol{x}(t) \\ \hat{\boldsymbol{x}}|_{t_0} &\equiv \boldsymbol{x}(t_0), & \hat{\boldsymbol{x}}|_{t_f} &\equiv \boldsymbol{x}(t_f) \end{aligned} \tag{8}$$

and $\boldsymbol{\alpha}$ is Lagrange multiplier defined only at the initial and final time.

Necessary conditions for optimality are derived by requiring that J be stationary with respect to variations in variables. Note that for the free final time problem, the following relations are given:

$$\delta\boldsymbol{x}(t_f) = \mathrm{d}\boldsymbol{x}(t_f) - \dot{\boldsymbol{x}}|_{t_f}\, \mathrm{d}t_f \qquad \delta\boldsymbol{\lambda}(t_f) = \mathrm{d}\boldsymbol{\lambda}(t_f) - \dot{\boldsymbol{\lambda}}|_{t_f}\, \mathrm{d}t_f \tag{9}$$

where $\delta\boldsymbol{x}(t_f)$ and $\delta\boldsymbol{\lambda}(t_f)$ denote the variations of state and costate vectors at fixed final time and $\mathrm{d}\boldsymbol{x}(t_f)$ and $\mathrm{d}\boldsymbol{\lambda}(t_f)$ denote the variations of state and costate vectors at free final time, respectively. The following variables are introduced for notational convenience:

$$\hat{\boldsymbol{\lambda}}_{t_0} \equiv \left(\frac{\partial\Phi}{\partial\boldsymbol{x}}\right)^T\Bigg|_{t_0}, \qquad \hat{\boldsymbol{\lambda}}_{t_f} \equiv \left(\frac{\partial\Phi}{\partial\boldsymbol{x}}\right)^T\Bigg|_{t_f} \tag{10}$$

The first variation of J is obtained after integrating by parts and manipulations as follows:

$$\begin{aligned} \delta J = \int_{t_0}^{t_f} \Bigg\{ & \delta\boldsymbol{\lambda}^T(\boldsymbol{f} - \dot{\boldsymbol{x}}) + \delta\boldsymbol{\beta}^T\boldsymbol{S} \\ & + \delta\boldsymbol{x}^T\left[\dot{\boldsymbol{\lambda}} + \left(\frac{\partial L}{\partial\boldsymbol{x}}\right)^T + \left(\frac{\partial\boldsymbol{f}}{\partial\boldsymbol{x}}\right)^T\boldsymbol{\lambda} + \left(\frac{\partial\boldsymbol{S}}{\partial\boldsymbol{x}}\right)^T\boldsymbol{\beta}\right] \\ & + \delta\boldsymbol{u}^T\left[\left(\frac{\partial L}{\partial\boldsymbol{u}}\right)^T + \left(\frac{\partial\boldsymbol{f}}{\partial\boldsymbol{u}}\right)^T\boldsymbol{\lambda}\right]\Bigg\}\,\mathrm{d}t \\ & + \delta\boldsymbol{\nu}^T\psi\big|_{t_0}^{t_f} + \mathrm{d}x^T(\hat{\boldsymbol{\lambda}} - \boldsymbol{\lambda})\big|_{t_0}^{t_f} + \delta\boldsymbol{\alpha}^T(\boldsymbol{x} - \hat{\boldsymbol{x}})\big|_{t_0}^{t_f} \\ & + \mathrm{d}t_f\left[L + \boldsymbol{\lambda}^T\boldsymbol{f} + \boldsymbol{\beta}^T\boldsymbol{S} + \frac{\partial\Phi}{\partial t}\right]_{t_f} \end{aligned} \tag{11}$$

Note that $(\mathrm{d}\boldsymbol{x} - \mathrm{d}\hat{\boldsymbol{x}})|_{t_0}^{t_f}$ may be set to zero since the admissible variation of state must be continuous at the initial and final times. Also, $\delta\boldsymbol{x}|_{t_0} = \mathrm{d}\boldsymbol{x}|_{t_0}$ since initial time is specified.

All of the coefficients of the variational terms must be zero for stationary values of J. By setting the coefficients of the $\delta\boldsymbol{x}^T$ and $\delta\boldsymbol{u}^T$ terms to zero, we can obtain the same Euler–Lagrange equation and optimality condition on the constrained arc in the case of state inequality constraint.[14,15] No jump condition of Lagrange multiplier is mentioned because there is no switching structure under the state equality constraints, which are active all of the time. And we only consider the case of continuous path constraints.

In previous research,[15] time differentiation of the path constraint was used to reduce the number of equations, and the resulting equations become linearly independent. In this study, instead, a numerical method is pursued to solve the full equations, because the state constraint equations have an indirect effect on control variables through the system dynamics. Detailed discussion will be given with numerical examples.

We will now derive a mixed form of δJ, which does not contain the time derivatives of state and costate variables. As shown in Ref. 7, the Lagrange multiplier $\boldsymbol{\alpha}$ can be chosen as follows:

$$\delta\boldsymbol{\alpha}(t_0) = \delta\boldsymbol{\lambda}(t_0), \qquad \delta\boldsymbol{\alpha}(t_f) = \delta\boldsymbol{\lambda}(t_f) \tag{12}$$

Substituting Eqs. (9) into Eq. (11) and integrating by parts yield

$$\begin{aligned} \delta J = \int_{t_0}^{t_f} \Bigg\{ & \delta\dot{\boldsymbol{\lambda}}^T\boldsymbol{x} - \delta\dot{\boldsymbol{x}}^T\boldsymbol{\lambda} + \delta\boldsymbol{\lambda}^T\boldsymbol{f} + \delta\boldsymbol{\beta}^T\boldsymbol{S} \\ & + \delta\boldsymbol{x}^T\left[\left(\frac{\partial L}{\partial\boldsymbol{x}}\right)^T + \left(\frac{\partial\boldsymbol{f}}{\partial\boldsymbol{x}}\right)^T\boldsymbol{\lambda} + \left(\frac{\partial\boldsymbol{S}}{\partial\boldsymbol{x}}\right)^T\boldsymbol{\beta}\right] \\ & + \delta\boldsymbol{u}^T\left[\left(\frac{\partial L}{\partial\boldsymbol{u}}\right)^T + \left(\frac{\partial\boldsymbol{f}}{\partial\boldsymbol{u}}\right)^T\boldsymbol{\lambda}\right]\Bigg\}\,\mathrm{d}t + \delta\boldsymbol{\nu}^T\psi\big|_{t_0}^{t_f} + \delta\boldsymbol{x}^T\hat{\boldsymbol{\lambda}}\big|_{t_0}^{t_f} \\ & - \delta\boldsymbol{\lambda}^T\hat{\boldsymbol{x}}\big|_{t_0}^{t_f} + \mathrm{d}t_f\left[L + \boldsymbol{\lambda}^T\boldsymbol{f} + \boldsymbol{\beta}^T\boldsymbol{S} + \frac{\partial\Phi}{\partial t}\right]_{t_f} \\ & + [(\hat{\boldsymbol{\lambda}} - \boldsymbol{\lambda})^T\dot{\boldsymbol{x}}]_{t_f}\,\mathrm{d}t_f - [(\hat{\boldsymbol{x}} - \boldsymbol{x})^T\dot{\boldsymbol{\lambda}}]_{t_f}\,\mathrm{d}t_f \end{aligned} \tag{13}$$

Note that the final conditions on state and costate are natural, the last two terms are equal to zero. Also, the term $\hat{\boldsymbol{\lambda}}|_{t_0}$ can be treated

as an unknown instead of $\nu|_{t_0}$ and, therefore, the $\nu|_{t_0}$ term can be removed from Eq. (13). Finally, the first variation of J in mixed form can be obtained as follows:

$$\delta J = \int_{t_0}^{t_f} \left\{ \delta\dot{\lambda}^T x - \delta\dot{x}^T\lambda + \delta\lambda^T f + \delta\beta^T S + \delta x^T \left[\left(\frac{\partial L}{\partial x}\right)^T + \left(\frac{\partial f}{\partial x}\right)^T \lambda + \left(\frac{\partial S}{\partial x}\right)^T \beta \right] + \delta u^T \left[\left(\frac{\partial L}{\partial u}\right)^T + \left(\frac{\partial f}{\partial u}\right)^T \lambda \right] \right\} dt + \delta\nu^T \psi|_{t_f} + \delta x^T \hat{\lambda}|_{t_0}^{t_f} - \delta\lambda^T \hat{x}|_{t_0}^{t_f} + d t_f \left[L + \lambda^T f + \beta^T S + \frac{\partial \Phi}{\partial t} \right]_{t_f} \tag{14}$$

Note that the necessary conditions for optimality require that the first variation of J be zero for all possible variations, and Eq. (14) is the governing equation for the inverse problem. To solve this problem through the finite element method, a discretization procedure is used.

III. Finite Element Discretization

In this section, a finite element discretization procedure for solving the optimal control problem is introduced. Let the time interval from t_0 to t_f be broken into N finite elements. The nodal values of these time elements are t_i $(i = 1, 2, \ldots, N+1)$, where $t_1 = t_0$ and $t_{N+1} = t_f$. A nondimensional elemental time τ is defined as follows:

$$\tau = \frac{t - t_i}{t_{i+1} - t_i} = \frac{t - t_i}{\Delta t_i}, \qquad 0 \le \tau \le 1 \tag{15}$$

where Δt_i is the element size. Because there are time derivative terms $\delta\dot{x}$ and $\delta\dot{\lambda}$ in Eq. (14), Hodges and Bless[7] chose the simple linear shape functions for δx and $\delta\lambda$ and constant shape functions for all of the other variables in the discretization procedure. With these selections, $\delta\dot{x}$ and $\delta\dot{\lambda}$ can be approximated as follows within the ith finite element:

$$\delta\dot{x} = \frac{\delta x_{i+1} - \delta x_i}{\Delta t_i} \tag{16}$$

$$\delta\dot{\lambda} = \frac{\delta\lambda_{i+1} - \delta\lambda_i}{\Delta t_i} \tag{17}$$

For the inverse problem, evaluation of the control input is most important, and in this study the linear shape function is also taken for control variable u. In summary, linear shape functions are used for δx, $\delta\lambda$, δu, and u, and constant shape functions are used for all of the other variables;

$$\delta x = (1-\tau)\delta x_i + \tau\delta x_{i+1} \tag{18}$$

$$\delta\lambda = (1-\tau)\delta\lambda_i + \tau\delta\lambda_{i+1} \tag{19}$$

$$\delta u = (1-\tau)\delta u_i + \tau\delta u_{i+1} \tag{20}$$

$$\delta\beta = \delta\beta_i \tag{21}$$

$$x = \begin{cases} \hat{x}_i, & \tau = 0 \\ \bar{x}_i, & 0 < \tau < 1 \\ \hat{x}_{i+1}, & \tau = 1 \end{cases} \tag{22}$$

$$\lambda = \begin{cases} \hat{\lambda}_i, & \tau = 0 \\ \bar{\lambda}_i, & 0 < \tau < 1 \\ \hat{\lambda}_{i+1}, & \tau = 1 \end{cases} \tag{23}$$

$$u = u_i(1-\tau) + u_{i+1}\tau \tag{24}$$

$$\beta = \beta_i \tag{25}$$

For convenience, the nodal values of x and λ are defined as $\hat{x}_i$ and $\hat{\lambda}_i$, respectively, and the elemental values of x and λ are defined as $\bar{x}_i$ and $\bar{\lambda}_i$, respectively. Also, the elemental values of β are defined as β_i, and the nodal values of u are defined as u_i. After substituting Eqs. (16–25) into Eq. (14), the integration should be performed within each finite element. Because the linear shape function is used for control variable u in our formulation, it is not possible to do the analytic integration as in Ref. 7. Instead, the trapezoidal rule is used as follows:

$$\int_0^1 \delta\lambda_i^T f(\bar{x}_i, u_i(1-\tau) + u_{i+1}\tau, t_i(1-\tau) + t_{i+1}\tau)\Delta t_i \, d\tau \approx \delta\lambda_i^T \frac{\Delta t_i}{2}\{f(\bar{x}_i, u_i, t_i) + f(\bar{x}_i, u_{i+1}, t_{i+1})\} \tag{26}$$

$$\int_0^1 \delta\lambda_{i+1}^T \tau f(\bar{x}_i, u_i(1-\tau) + u_{i+1}\tau, t_i(1-\tau) + t_{i+1}\tau)\Delta t_i \, d\tau \approx \delta\lambda_{i+1}^T \frac{\Delta t_i}{2} f(\bar{x}_i, u_{i+1}, t_{i+1}) \tag{27}$$

Now, the governing equation, Eq. (14), can be discretized as in the following algebraic form:

$$\sum_{i=1}^{N} \Bigg\langle \delta x_i^T \left[\bar{\lambda}_i + \frac{\Delta t_i}{2} \left\{ \left(\frac{\partial L}{\partial x}\right)_i + \left(\frac{\partial f}{\partial x}\right)_i^T \bar{\lambda}_i + \left(\frac{\partial S}{\partial x}\right)_i^T \beta_i \right\} \right] + \delta x_{i+1}^T \left[-\bar{\lambda}_i + \frac{\Delta t_i}{2} \left\{ \left(\frac{\partial L}{\partial x}\right)_{i+1} + \left(\frac{\partial f}{\partial x}\right)_{i+1}^T \bar{\lambda}_i + \left(\frac{\partial S}{\partial x}\right)_{i+1}^T \beta_i \right\} \right] + \delta\lambda_i^T \left[-\bar{x}_i + \frac{\Delta t_i}{2} f_i \right] + \delta\lambda_{i+1}^T \left[\bar{x}_i + \frac{\Delta t_i}{2} f_{i+1} \right] + \delta\beta_i^T \left[\frac{\Delta t_i}{2}(S_i + S_{i+1}) \right] + \delta u_i^T \left[\frac{\Delta t_i}{2} \left\{ \left(\frac{\partial L}{\partial u}\right)_i + \left(\frac{\partial f}{\partial u}\right)_i^T \bar{\lambda}_i \right\} \right] + \delta u_{i+1}^T \left[\frac{\Delta t_i}{2} \left\{ \left(\frac{\partial L}{\partial u}\right)_{i+1} + \left(\frac{\partial f}{\partial u}\right)_{i+1}^T \bar{\lambda}_i \right\} \right] \Bigg\rangle - \delta x_1^T \hat{\lambda}_1 + \delta x_{N+1}^T \hat{\lambda}_{N+1} + \delta\lambda_1^T \hat{x}_1 - \delta\lambda_{N+1}^T \hat{x}_{N+1} + \delta\nu^T \psi|_{t_{N+1}} + d t_f \left[L + \lambda^T \hat{f} + \beta^T S + \frac{\partial \Phi}{\partial t} \right]_{t_{N+1}} = 0 \tag{28}$$

where

$$f_i = f(\bar{x}_i, u_i, t_i) \qquad f_{i+1} = f(\bar{x}_i, u_{i+1}, t_{i+1})$$

$$\left(\frac{\partial L}{\partial x}\right)_i = \frac{\partial L}{\partial x}(\bar{x}_i, u_i, t_i) \qquad \left(\frac{\partial L}{\partial x}\right)_{i+1} = \frac{\partial L}{\partial x}(\bar{x}_i, u_{i+1}, t_{i+1})$$

$$L|_{t_{N+1}} = L(\hat{x}_{N+1}, u_{N+1}, t_{N+1}) \tag{29}$$

and the other notations follow the same conventions.

By setting the coefficients of all variational terms in Eq. (28) equal to zero, we obtain a set of nonlinear algebraic equations that is the resulting discretized governing equations for the inverse problem. Because the discretized governing equations are derived before the problem is specified, these equations are in a general form and may be encoded into a general program. Solving these equations, one obtains optimized numerical solutions to any inverse problem for a specific dynamic system, performance index, and specified various constraints.

The procedure for obtaining a numerical solution is now explained in detail. Elemental variables such as $\bar{\boldsymbol{x}}_i$, $\bar{\boldsymbol{\lambda}}_i$, and $\bar{\boldsymbol{u}}_i$ are evaluated by solving the algebraic equations. Then nodal values $\hat{\boldsymbol{x}}_i$ and $\hat{\boldsymbol{\lambda}}_i$ can be computed by using the following recursive relation[7]:

$$\bar{\boldsymbol{x}}_i = \frac{(\hat{\boldsymbol{x}}_i + \hat{\boldsymbol{x}}_{i+1})}{2} \tag{30}$$

$$\bar{\boldsymbol{\lambda}}_i = \frac{(\hat{\boldsymbol{\lambda}}_i + \hat{\boldsymbol{\lambda}}_{i+1})}{2} \tag{31}$$

Nodal control value $\hat{\boldsymbol{u}}_i$ can be evaluated by applying the optimality condition with the nodal values $\hat{\boldsymbol{x}}_i$ and $\hat{\boldsymbol{\lambda}}_i$.

There are $(2n + m)(N + 1) + lN + q + 1$ nonlinear algebraic equations and the same number of unknowns: $\bar{\boldsymbol{x}}_i$, $\bar{\boldsymbol{\lambda}}_i$, β_i $(i = 1, 2, \ldots, N)$, and $\boldsymbol{u}_i$ $(i = 1, 2, \ldots, N + 1)$, and $\hat{\boldsymbol{x}}_{N+1}$, $\hat{\boldsymbol{\lambda}}_i$, $\boldsymbol{\nu}$, and t_f. Because the nodal control values $\boldsymbol{u}_i$ appear explicitly in the discretized governing equations, no additional procedures are necessary to evaluate $\boldsymbol{u}_i$. Thus, error propagation due to control nodal value calculations is avoided. Also, less computational time, fewer possibilities of ill-conditioning, and better numerical robustness are expected from the fact that no numerical differentiation and/or integration procedures are required in the proposed algorithm.

Some of the discretized governing equations become linearly dependent on each other and, also, some Lagrange multipliers may be determined by implicit relations. This leads to the Jacobian matrix being singular, and therefore these Lagrange multipliers may be calculated utilizing the pseudoinverse concept. In this study, however, the noncontributing Lagrange multipliers can be identified through the Jacobian matrix that is provided explicitly, and therefore the numerical results satisfy the conditions for optimality.

The Levenberg–Marquardt method,[16] a least square optimization algorithm, is adopted to solve the problem numerically. A symbolic differentiation program is used to derive the analytic formulation of the Jacobian matrix, $\partial \boldsymbol{f}/\partial \boldsymbol{x}$, and $\partial L/\partial \boldsymbol{u}$, in Eq. (28), and a sparse linear solver is used to exploit the sparsity of the Jacobian matrix. To solve the nonlinear algebraic equations, an initial guess for each unknown is needed at each iteration. In this study, the first initial guesses for state variables are chosen by linear interpolation of the given initial conditions and the desired final conditions, and the initial guesses for all Lagrange multipliers are set to zero with two or four time elements. In the following iterations, the preceding solution is linearly interpolated to construct the next iterative's initial guess to get the refined solution. Final solutions are obtained by increasing the element number and repeating this procedure until the required accuracy is obtained.

IV. Numerical Examples

A relatively simple nonlinear model is chosen to compare with previous work.[3,4] Assume that the forward velocity U is constant and the same as the total velocity V_t $(U = V_t)$. Using linearized kinematics and aerodynamic forces,[17] the equations of rigid aircraft motion are given by

$$\dot{Y} = V_t \Psi + V \cos\Phi - W \sin\Phi \tag{32}$$

$$\dot{Z} = -V_t \Theta + V \sin\Phi + W \cos\Phi \tag{33}$$

$$\dot{V} = -V_t R + PW + Y_V V + Y_{\delta_R} \delta_R + g \sin\Phi \tag{34}$$

$$\dot{W} = V_t Q - PV + Z_W W + Z_{\delta_E} \delta_E + g \cos\Phi \tag{35}$$

$$\dot{\Phi} = P \tag{36}$$

$$\dot{\Theta} = Q \cos\Phi - R \sin\Phi \tag{37}$$

$$\dot{\Psi} = Q \sin\Phi + R \cos\Phi \tag{38}$$

$$\dot{P} = L'_V V + L'_R R + L'_P P + L'_{\delta_R} \delta_R + L'_{\delta_A} \delta_A \tag{39}$$

$$\dot{Q} = -\mu PR + M_W W + M_Q Q + M_{\delta_E} \delta_E \tag{40}$$

$$\dot{R} = \sigma PQ + N'_V V + N'_R R + N'_P P + N'_{\delta_R} \delta_R + N'_{\delta_A} \delta_A \tag{41}$$

where $\mu = (I_x - I_z)/I_y$ and $\sigma = (I_y - I_x)/I_z$. Here, 10 state variables define positions in the inertial coordinates Y and Z; body axis linear velocities V and W; standard Euler angles Φ, Θ, and Ψ; and angular velocities P, Q, and R. Three control inputs, δ_R, δ_E, and δ_A, denote rudder, elevator, and aileron deflections, respectively. Trim conditions, $\alpha_0 = 3.8288$ deg (0.0668 rad) and $\delta_{E_0} = -2.5038$ deg (−0.0436 rad), in straight level flight at an altitude of 4.6 km (15,000 ft) and at a velocity of $M = 0.6$ (634 ft/s), are calculated and used as the initial conditions.

Aileron Roll Maneuver

In aileron roll maneuver, the aircraft sustains a straight flight path and rolls through 360 deg in T s. Desired path constraints can be expressed as[3,5]

$$Y = 0 \tag{42}$$

$$Z = 0 \tag{43}$$

$$\Phi = (2\pi/16)[\cos(3\pi t/T) - 9\cos(\pi t/T) + 8] \tag{44}$$

This is the nominal case with three control inputs and three path constraints, and so there is no freedom to optimize any performance index. But a virtual optimization problem can be formulated to manipulate the inverse problem using the general algorithm developed in this paper. To minimize the control surface deflections and to reattain the initial trim condition after 360-deg roll, the performance index and the final time constraint are set as follows:

$$L = \tfrac{1}{2}\left(\delta_R^2 + \delta_E^2 + \delta_A^2\right) \tag{45}$$

$$\boldsymbol{\psi} = [Y \;\; Z \;\; V \;\; (W - W_0) \;\; \Phi \;\; (\Theta - \Theta_0) \;\; \Psi \;\; P \;\; Q \;\; R]^T \tag{46}$$

where W_0 is the initial vertical velocity and Θ_0 is the initial pitch angle. With these path constraints and boundary conditions, unrealistic results similar to those in Kato and Sugiura[3] are found. The results are designated by lines marked with triangles and plus signs in Fig. 1.

The results marked with triangles are obtained by the procedure devised in this paper, and the results marked with plus signs are obtained by the differential approach explained in Ref. 18. These two results show some discrepancy, presumably caused by differences in numerical properties between the two approaches, but can be considered as acceptable engineering solutions for the given inverse problem. We can say that the direct augmentation method, in which no explicit time differentiation of path constraint is needed, can find a numerical control input for the inverse problem. All control surface deflections are excessive, and the sideslip angle varies from −20 to +20 deg. Note that roll angle follows the prescribed trajectory well, and the Y–Z trajectory deviations from the straight flight path are less than 10 ft.

To get a more realistic maneuver, we impose only one constraint on the roll angle Φ, as in Eq. (44), and use the same final time constraint. The new results represented in Fig. 1 (circle and cross marks) show the good bank angle match, reduced control surface deflections, well coordinated sideslip angle response, and reasonable Y–Z trajectory. These results are better than those of both Kato and Sugiura[3] and Sentoh and Bryson.[4]

The high-performance fighters developed recently have more control redundancies such as unconventional aerodynamic control surfaces and thrust vectoring capability. Therefore, the methodology presented can be used to design the control law for a fighter aircraft with high-control redundancy. Many problems involving the trajectory optimization with the consideration of dynamics in robotic engineering can be solved as well.

Bank-to-Bank Maneuver

For bank-to-bank maneuver, the aircraft sustains zero flight-path angle and zero yaw angle and follows the given sinusoidal bank angle trajectory. Gao and Hess[5] formulated this maneuver as

$$\Theta = \alpha \tag{47}$$

$$\Psi = 0 \tag{48}$$

$$\Phi = \Phi_M \sin(\Omega_r t) \tag{49}$$

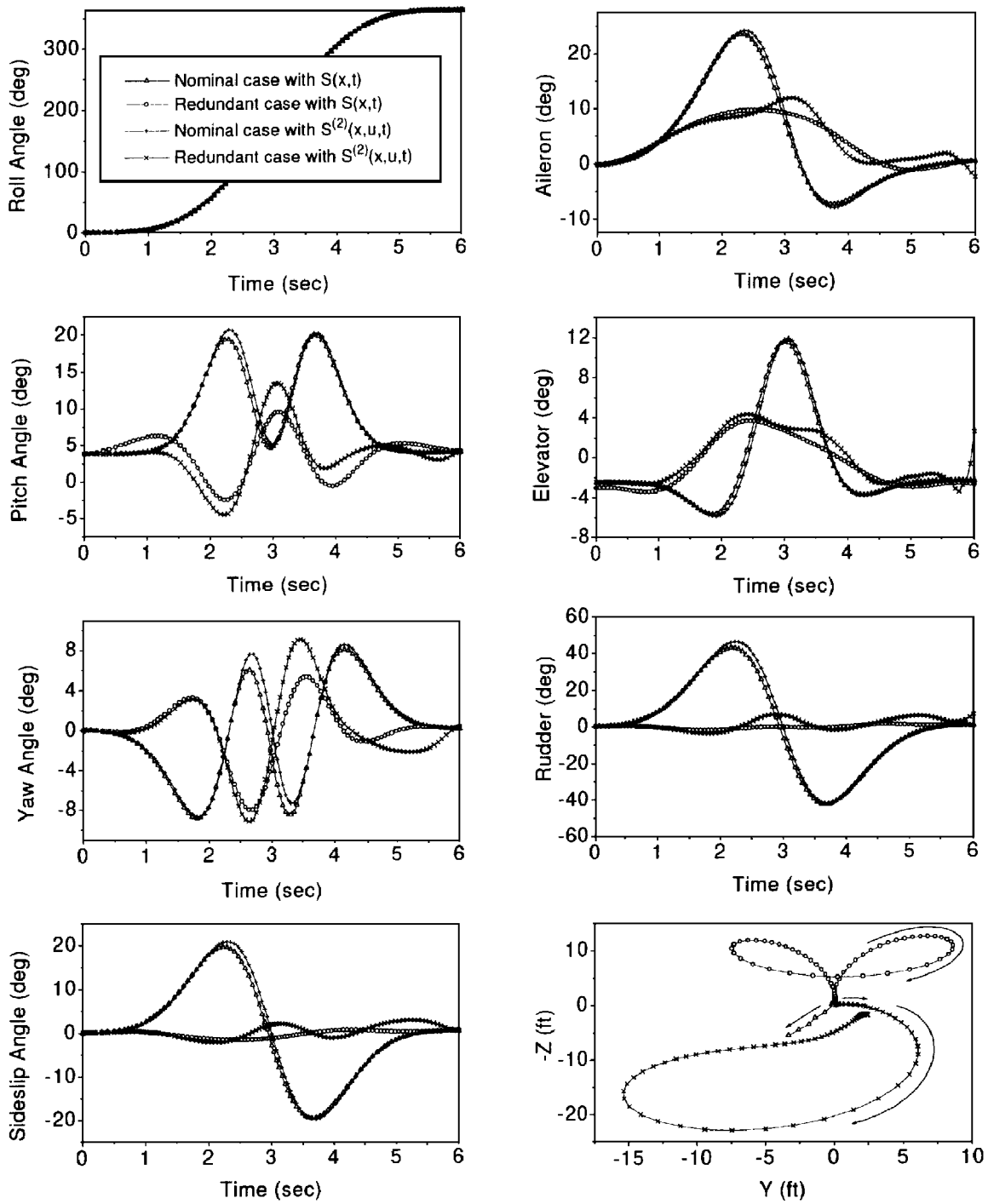

Fig. 1 Aileron roll maneuver.

where Φ_M denotes the maximum bank angle and Ω_r determines the frequency of the desired sinusoidal profile. According to the approximation used in the modeling procedure, the $\Theta = \alpha$ constraint may be approximated as $\Theta = W/V_t$.

The performance index is set as $L = (\delta_R^2 + \delta_E^2 + \delta_A^2)/2$, which minimizes the control surface deflections, and the final time constraints of Eq. (46) are also adopted to reattain the initial trim condition. This inverse problem is solved by the presented method with the numerical values of $\Phi_M = 10$ deg, $\Omega_r = 0.1$ Hz, and $t_f = 20$ s. Severe oscillations of control inputs are obtained, similar to Gao and Hess,[5] as shown in Fig. 2 (triangle mark).

Gao and Hess[5] explained that a local optimal solution or numerical error might be causing the control oscillations and, therefore, low-pass filtering techniques could be used to smooth the control inputs. However, in our opinion the cause of control oscillations comes from the incompatibility of the given path constraints with the initial condition. At trim condition the roll rate P is zero, but the given path constraint demands a value of $\Phi_M \Omega_r$ deg/s.

A new path constraint equation is conceived such that 1) it is compatible with the initial condition and 2) it properly describes the desired bank-to-bank maneuver. From 0 to 2.5 s and 17.5 to 20 s, the new path constraint equation consists of a seventh-order polynomial, and between 2.5 and 17.5 s, the original sinusoidal equation of Eq. (49) is used.

Using this new path constraint, good bank angle following is achieved with smooth control input history, as shown in Fig. 2 (circles). In both cases, yaw angle is not exactly sustained at 0 deg; however, the deviation is within ± 0.05 deg, which is acceptable in real-flight situations. This example confirms that the design of the proper path constraint is not a trivial procedure.

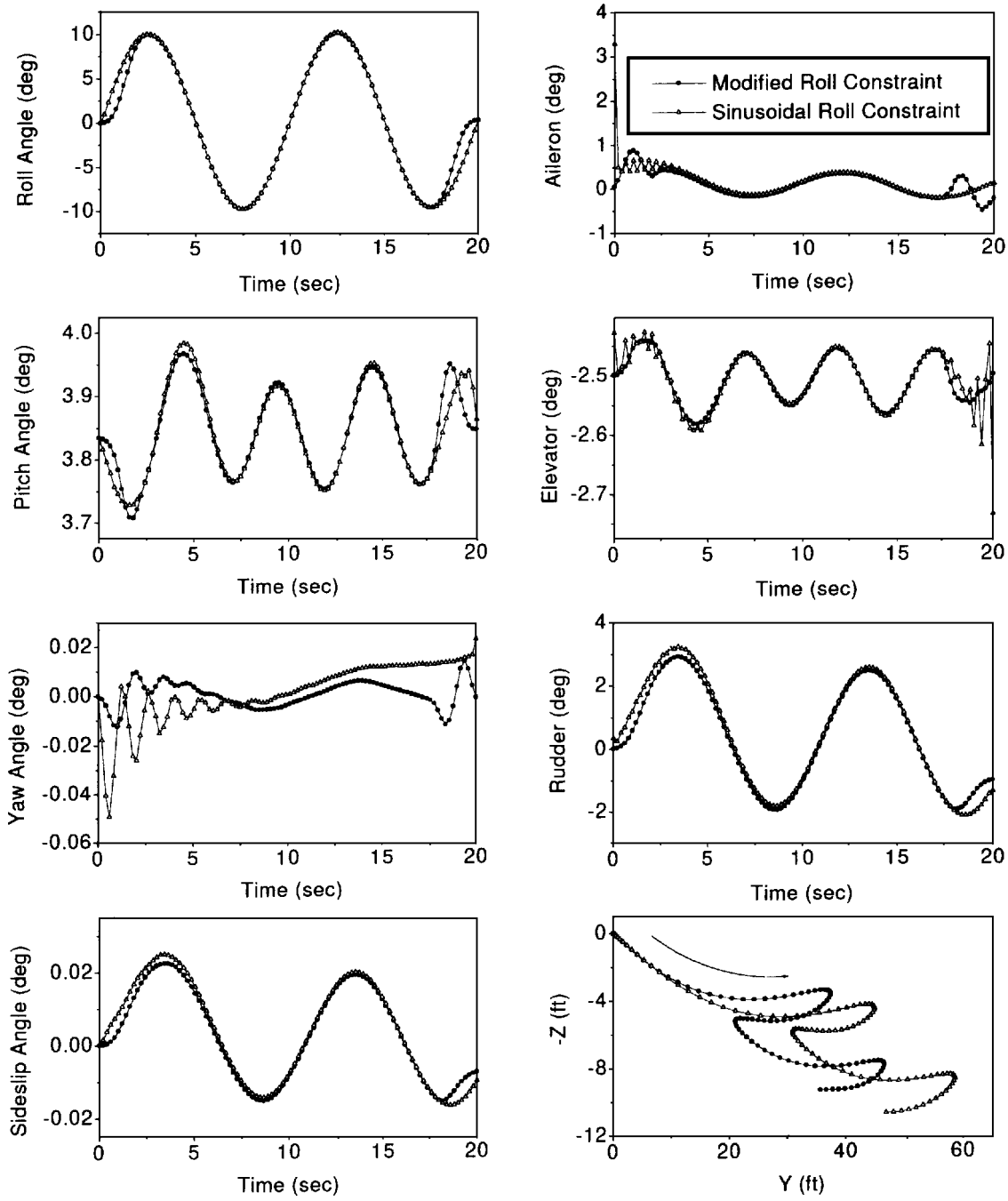

Fig. 2 Bank-to-bank maneuver.

V. Conclusion

A new method solving the inverse problem is introduced by formulating the inverse problem as a general optimization problem with equality constraints that are functions of the state variables. The optimality conditions are derived using a variational approach based on Hamilton's weak principle. The time-domain finite element method is used to discretize the derived governing equations, and a numerical solution is obtained through least square optimization. The proposed algorithm can fully utilize the control redundancy for the redundant case through proper selection of the performance index and constraints. Time differentiation of trajectory constraints and partial differentiation of output variables with respect to the control inputs are not required. To verify the effectiveness of the proposed method, solutions for an aileron roll maneuver and a bank-to-bank maneuver of aircraft are presented.

Acknowledgment

This work is supported by a research fund from the Korea Science and Engineering Foundation (Grant 951-1003-008-1).

References

[1]Lu, P., "Inverse Dynamics Approach to Trajectory Optimization for an Aerospace Plane," *Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 4, 1993, pp. 726–732.

[2]Lu, P., and Pierson, B. L., "Optimal Aircraft Terrain-Following Analysis and Trajectory Generation," *Journal of Guidance, Control, and Dynamics*, Vol. 18, No. 3, 1995, pp. 555–560.

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

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

[5]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.

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

[7]Hodges, D. H., and Bless, R. R., "Weak Hamiltonian Finite Element Method for Optimal Control Problems," *Journal of Guidance, Control, and Dynamics*, Vol. 14, No. 1, 1991, pp. 148–156.

[8]Hodges, D. H., and Bless, R. R., "Finite Element Solution of Optimal Control Problems with State-Control Inequality Constraints," *Journal of Guidance, Control, and Dynamics*, Vol. 15, No. 4, 1992, pp. 1029–1032.

[9]Bless, R. R., "Time-Domain Finite Elements in Optimal Control with Application to Launch-Vehicle Guidance," NASA CR-4376, 1991.

[10]Hodges, D. H., Bless, R. R., Calise, A. J., and Leung, M., "Finite Element Method for Optimal Guidance of an Advanced Launch Vehicle," *Journal of Guidance, Control, and Dynamics*, Vol. 15, No. 3, 1992, pp. 664–671.

[11]Bless, R. R., Hodges, D. H., and Seywald, H., "State-Constrained Booster Trajectory Solutions via Finite Elements and Shooting," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Washington, DC, 1993, pp. 406–416 (AIAA Paper 93-3747).

[12]Bless, R. R., and Queen, E. M., "Variational Trajectory Optimization Tool Set," NASA TM-4442, July 1993.

[13]Bless, R. R., Hodges, D. H., and Seywald, H., "Finite Element Method for the Solution of State-Constrained Optimal Control Problems," *Journal of Guidance, Control, and Dynamics*, Vol. 18, No. 5, 1995, pp. 1036–1043.

[14]Bryson, A. E., Jr., Denham, W. F., and Dreyfus, S. E., "Optimal Programming Problems with Inequality Constraints I: Necessary Conditions for Extremal Solutions," *AIAA Journal*, Vol. 1, No. 11, 1963, pp. 2544–2550.

[15]Speyer, J. L., and Bryson, A. E., Jr., "Optimal Programming Problems with a Bounded State Space," *AIAA Journal*, Vol. 6, No. 8, 1968, pp. 1488–1491.

[16]Grace, A., *Optimization Toolbox User's Guide*, MathWorks, Inc., Natick, MA, 1992.

[17]McRuer, D., Ashikenas, I., and Graham, D., *Aircraft Dynamics and Automatic Control*, Princeton Univ. Press, Princeton, NJ, 1973.

[18]Bryson, A. E., Jr., and Ho, Y. C., *Applied Optimal Control*, Wiley, New York, 1975.