Development of Linear-Parameter-Varying Models for Aircraft

Andrés Marcos* and Gary J. Balas†
University of Minnesota, Minneapolis, Minnesota 55455

This paper presents a comparative study of three linear-parameter-varying (LPV) modeling approaches and their application to the longitudinal motion of a Boeing 747 series 100/200. The three approaches used to obtain the quasi-LPV models are Jacobian linearization, state transformation, and function substitution. Development of linear parameter varying models are a key step in applying LPV control synthesis. The models are obtained for the up-and-away flight envelope of the Boeing 747-100/200. Comparisons of the three models in terms of their advantages, drawbacks, and modeling difficulty are presented. Open-loop time responses show the three quasi-LPV models matching the behavior of the nonlinear model when in the trim region. Differences between the models are more apparent as the response of the aircraft deviates from the nominal trim conditions.

Nomenclature

$c_1, \ldots, c_9$	=	inertia coefficients
$\bar{c}$	=	wing chord, m
$\mathcal{F}_{\text{dec}}$	=	decomposition function
m	=	aircraft mass, kg
$\bar{q}$	=	dynamic pressure, N/m^2
S	=	reference surface area, m^2
s_α, c_α	=	sine, cosine AoA
$w(t)$	=	nonscheduling states
x, z_{cg}	=	aircraft center of gravity position x, z axis, m
$z(t)$	=	scheduling states
z_{eng}	=	engine position z axis, m
α	=	angle of attack (AoA), deg
α_w	=	wing design plane $\alpha_w = \alpha + 2$ deg
$\rho(t)$	=	scheduling vector

I. Introduction

IN the last decade linear-parameter-varying (LPV) control has been presented as a reliable alternative to classical gain scheduling for multivariable systems. Gain scheduling is a standard method to design controllers for dynamical systems over a wide performance envelope. It yields a global controller based on interpolation of a family of locally linearized controllers. Drawbacks of this methodology are its ad hoc character and more importantly the fact that the controller obtained comes with no guarantees on its stability or performance other than at the design points (this is especially critical for rapid variations in the scheduling parameters). Linear-parameter-varying synthesis techniques naturally fit into the gain-scheduling framework, while imbuing it with local stability and performance assurances. LPV control synthesis techniques have already been used, with varying levels of success, for a wide array of dynamical systems. These include high-performance aircraft as representative as the F-14[1], F-16[2], F-18[3], and the VAAC Harrier;[4] turbofan engines;[5,6] and missiles;[7,8] and, recently, reconfigurable controllers.[9,10] A condition to apply LPV control synthesis is to transform the nonlinear model of the system into an LPV model;

hence, LPV modeling becomes a key issue in the design of LPV controllers.[4,11,12] Unfortunately, LPV modeling is not frequently considered as an independent research topic. Most of the available literature is only concerned with LPV control and either assumes a given LPV model or the model obtained is only of interest for the particular application being studied. Recently, two survey papers on the area of gain scheduling have appeared that treat the subject of LPV and quasi-LPV (qLPV) modeling and present detailed analysis of some of the available methods (see Refs. 13 and 14). A qLPV model is a particular case of an LPV model, which is characterized by having a subset of the scheduling parameters belong to the state space of the system.

Generally, control designers use a family of linear, time-invariant (LTI) plants at different points of interest throughout the operational envelope in order to obtain an LPV model.[1,3,5,6] This is known as the Jacobian linearization approach. It is based on first-order linear approximations with respect to a set of equilibrium points. In Refs. 4, 11, and 12, state transformations techniques were used to derive a reliable LPV model. The state transformation approach performs a state transformation based on differentiable functions of the nonscheduling states and control inputs with the goal of removing any nonlinear terms not dependent on the scheduling parameter. A third LPV modeling approach, function substitution, proposes a substitution of a so-called decomposition function $\mathcal{F}_{\text{dec}}$ by a linear combination of (scheduling parameter-dependent) functions and the scheduling vector. This decomposition function is formed by combining all of the terms of the nonlinear system that are not both affine with respect to the nonscheduling states and control inputs, and function of the scheduling vector alone. It has been used recently to model the F-16 aircraft dynamics for LPV and receding horizon control.[2,15] Other modeling approaches involve the use of linear fractional transformations, velocity-based approaches, or different types of linearizations (see Refs. 13, 14, and 16).

In this paper, the first three approaches—Jacobian linearization, state transformation, and function substitution—are used to model the Boeing 747-100/200 longitudinal motion in the up-and-away flight envelope. A theoretical examination of the modeling approaches is presented first. The nonlinear aircraft models, a high- and a low-fidelity model, and the software used, FTLAB747v61/65, are aintroduced next. The low-fidelity model, basically a reduction of the stability derivatives, is used for the qLPV modeling stage to facilitate the modeling task. A discussion on the qLPV model development and on their respective advantages and disadvantages follows. To validate the qLPV models, open-loop time responses are obtained and compared to the nonlinear aircraft responses. The results show that for a specific region in the flight envelope studied, the acceptable trim region, all of the three qLPV are valid, but that for excursions away from this region the function substitution qLPV model tends to give a better match.

*Graduate Student, Department of Aerospace Engineering and Mechanics, 110 Union Street S.E.; marcosa@aem.umn.edu.
†Professor, Department of Aerospace Engineering and Mechanics; balas@aem.umn.edu.

II. Theory

This section introduces background theory and definitions of LPV and quasi-LPV systems. The class of finite dimensional linear systems whose state-space entries depend continuously on a time-varying parameter vector $\rho(t)$ is called linear parameter varying. The trajectory of the vector valued signal $\rho(t)$ is assumed not to be known in advance, although its value can be accessed (measured) in real time and is constrained a priori to lie in a specified bounded set. The idea behind using LPV systems in lieu of LTI or linear time varying is to take advantage of the causal knowledge of the dynamics of the system. In the LPV framework, this causal relationship between the vector value signal $\rho(t)$ and the plant allows the control designer to restrict the dependence of the controller dynamics to variations in the plant characteristics. Hence, one is able to take full advantage of the information provided by the scheduling variables. The definition of LPV systems follows.

Definition 1 Linear Parameter Varying Systems[17]*:* Given a compact subset $\mathcal{P} \subset \mathcal{R}^s$, the parameter variation set $\mathcal{F}_\mathcal{P}$ denotes the set of all piecewise continuous functions mapping $\mathcal{R}^+$ (time) into $\mathcal{P}$ with a finite number of discontinuities in any interval. Given continuous functions, $A : \mathcal{R}^s \to \mathcal{R}^{n \times n}$, $B : \mathcal{R}^s \to \mathcal{R}^{n \times n_u}$, $C : \mathcal{R}^s \to \mathcal{R}^{n_y \times n}$, and $D : \mathcal{R}^s \to \mathcal{R}^{n_y \times n_u}$.

A nth-order linear parameter-varying system is defined as

$$\begin{bmatrix} \dot{x}(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} A(\rho(t)) & B(\rho(t)) \\ C(\rho(t)) & D(\rho(t)) \end{bmatrix} \begin{bmatrix} x(t) \\ u(t) \end{bmatrix} \quad (1)$$

where $\rho(t) \in \mathcal{F}_\mathcal{P}$.

Quasi-LPV systems arise when a scheduling variable $\rho(t)$ is also a state of the system. This state dependency might result in a nonlinear feedback through the system matrices, and hence the term quasi is more appropriately used to define this particular class of systems. By treating the scheduling parameters as independent variables (at the expense of possibly some conservativeness), the techniques used to design LPV controllers can be applied to qLPV models.[7]

Definition 2 (Quasi-Linear-Parameter-Varying Systems): Given a linear-parameter-varying system as defined in definition 1, a quasi-linear-parameter-varying system is obtained if the state vector $x(t)$ can be decomposed into scheduling states $z(t) \in \mathcal{F}_\mathcal{P}$ and nonscheduling states $w(t)$.

$$x(t) = [z(t) \quad w(t)]^T \quad (2)$$

Thus, the quasi-LPV model is defined by

$$\begin{bmatrix} \dot{z}(t) \\ \dot{w}(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} A_{11}(\rho(t)) & A_{12}(\rho(t)) & B_1(\rho(t)) \\ A_{21}(\rho(t)) & A_{22}(\rho(t)) & B_2(\rho(t)) \\ C_1(\rho(t)) & C_2(\rho(t)) & D(\rho(t)) \end{bmatrix} \begin{bmatrix} z(t) \\ w(t) \\ u(t) \end{bmatrix} \quad (3)$$

where the scheduling parameter vector is $\rho(t) = [z(t) \ \Omega(t)]$ and $\Omega(t) \in \mathcal{R}^{n_p}$ are exogenous scheduling variables.

The selection of scheduling variables that capture the nonlinearities of the system is a task that is not always obvious a priori. There are several approaches that can be used to obtain a reliable LPV/quasi-LPV model.[2,13,14,18] This paper focuses on three techniques: two of them well-known, Jacobian linearization and state transformation, and one of recent appearance, function substitution. Assume that the nonlinear model is of the following class:

$$\begin{bmatrix} \dot{z}(t) \\ \dot{w}(t) \end{bmatrix} = A(\rho(t)) \begin{bmatrix} z(t) \\ w(t) \end{bmatrix} + B(\rho(t))[u(t)] + K(\rho(t)) \quad (4)$$

$$y(t) = [z(t) \quad \Omega(t)]^\top \quad (5)$$

where $z(t) \in \mathcal{R}^{n_z}$ is the scheduling-states vector, $w(t) \in \mathcal{R}^{n_w}$ the nonscheduling states, $u(t) \in \mathcal{R}^{n_u}$ is the control input vector, and the measured output vector is given by $y(t) \in \mathcal{R}^{n_y}$. The states and the control inputs must enter the system in a linear fashion. In princi-

ple, there might be relaxations on the linearity of any variable by considering it a scheduling parameter and hence inside the $K(\rho(t))$ matrix. An example of this type of linear relaxation can be found in Ref. 19. There are no restrictions with respect to the dependency of the A, B, and K matrices on the scheduling vector $\rho(t)$. Without loss of generality, it is assumed that there are no exogenous scheduling variables $[\rho(t) = z(t)]$. From this type of system, it is possible to develop each of the qLPV models, and hence a common departure point is provided to enable a more reliable comparison. This class of nonlinear system is sufficiently general; most of the aerospace systems currently used can be cast in this form with minor assumptions. The dependency on time will be dropped from the equations in the sequel.

A. Jacobian Linearization

The Jacobian linearization approach is the most widespread methodology used to obtain LPV models. It is also applicable to the widest class of nonlinear systems. It is valid for any nonlinear system that can be linearized at its equilibrium points, and therefore covers a much richer nonlinear class than the model given in Eq. (4). It can be used to create an LPV system based on a family of plants linearized with respect to a set of equilibrium points that represents the flight envelope of interest. The resulting model is a local approximation to the dynamics of the nonlinear plant around this set of equilibrium points. The basis of this method is to use a first-order Taylor-series expansion of the nonlinear model, Eq. (4), with respect to a trim point, $\delta_{f_1} = f_1(z, w, u) - f_1(z, w, u)|_{eq}$:

$$\delta_{f_1} = \Delta_z f \cdot \delta_z + \Delta_w f \cdot \delta_w + \Delta_u f \cdot \delta_u$$

$$= \Delta_z f \cdot (z - z_{eq}) + \Delta_w f \cdot (w - w_{eq}) + \Delta_u f \cdot (u - u_{eq}) \quad (6)$$

The term $\Delta_z f$ indicates derivative of the function f with respect to the variable z. Rewrite the resulting equations for the different class of states in state-space form as follows:

$$\begin{bmatrix} \dot{\delta}_z \\ \dot{\delta}_w \end{bmatrix} = \begin{bmatrix} \Delta_z f_1 & \Delta_w f_1 \\ \Delta_z f_2 & \Delta_w f_2 \end{bmatrix}\bigg|_{eq} \begin{bmatrix} \delta_z \\ \delta_w \end{bmatrix} + \begin{bmatrix} \Delta_u f_1 \\ \Delta_u f_2 \end{bmatrix}\bigg|_{eq} [\delta_u] \quad (7)$$

It is easy to verify that the trim values and all of the elements in the state-space matrices depend on the scheduling variables, and hence the model is quasi-LPV. A more detailed theoretical derivation of a Jacobian model is given in Ref. 18.

Because it is a first-order approximation, it could lead to divergent behavior (with respect to the nonlinear model) for large control inputs. The local nature of the system could be improved by using higher-order terms in the Taylor-series expansion, but this could lead to impractical implementations.[16] In Ref. 16 it was also pointed out that a q/LPV model is conceptually different to a family of linearized systems because the latter is a collection of dynamical systems defined by perturbations and the former is a single dynamic system. It is generally impossible to capture the transient behavior of the nonlinear plant by this method, although for a certain class of nonlinear systems it is possible to account for its essential features.[12]

B. State Transformation

The second approach is called state transformation because the quasi-LPV model is obtained through exact transformations of the nonlinear states. It basically performs a state transformation based on differentiable functions of the nonscheduling states and control inputs with the goal of removing any nonlinear terms not dependent on the scheduling parameter. This technique was introduced by Shamma and Cloutier,[11] and it has been applied to a wide range of applications.[2,4,7,12] The application of this technique is restricted to the special class of nonlinear systems given in Eq. (4) (also known as an output nonlinear system). It is required that $n_z = n_u$ in order to transform Eq. (4) into a qLPV model whose state-space data are functions of the scheduling variables ρ. It is also required that the scheduling parameters are available in real time for measurement.

Assume there exist continuously differentiable functions $w_{eq}(\rho)$ and $u_{eq}(\rho)$ such that for every $\rho \in \mathcal{F}_{\mathcal{P}}$ the system is in steady state:

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} K_1(\rho) \\ K_2(\rho) \end{bmatrix} + \begin{bmatrix} A_{11}(\rho) & A_{12}(\rho) \\ A_{21}(\rho) & A_{22}(\rho) \end{bmatrix} \begin{bmatrix} z \\ w_{eq}(\rho) \end{bmatrix}$$

$$+ \begin{bmatrix} B_1(\rho) \\ B_2(\rho) \end{bmatrix} [u_{eq}(\rho)] \tag{8}$$

Unfortunately, there are no assurances about the existence of these trim functions or their validity within the feasible flight region. It is only possible to ensure that the model obtained is valid in the restricted envelope (that with acceptable trim functions). Consequently, before generating the qLPV model it is necessary to investigate the feasible trim map.

The trim functions defined previously, Eq. (8), can be used to transform the state definition in Eq. (4). Through some basic algebraic manipulations the following qLPV model is obtained:

$$\begin{bmatrix} \dot{z} \\ \dot{w} - \dot{w}_{eq} \end{bmatrix} = \begin{bmatrix} 0 & A_{12}(\rho) \\ 0 & A_{22}(\rho) - \left.\frac{\partial w_{eq}}{\partial z}\right|_\rho A_{12}(\rho) \end{bmatrix} \begin{bmatrix} z \\ w - w_{eq}(\rho) \end{bmatrix}$$

$$+ \begin{bmatrix} B_1(\rho) \\ B_2(\rho) - \left.\frac{\partial w_{eq}}{\partial z}\right|_\rho B_1(\rho) \end{bmatrix} [u - u_{eq}(\rho)] \tag{9}$$

Note that the nonlinear terms matrix $K(\rho)$ has now disappeared. The qLPV model represents the nonlinear system generated through an exact transformation. References 2, 11, 12, 16, 18, and 20 also provide more discussions and details about this approach.

C. Function Substitution

The function substitution approach was first proposed in Ref. 7 for qLPV systems with nonlinearities in the control input; recall that qLPV systems must be linear with respect to the nonscheduling states and control inputs. In that reference a transformation of the nonlinear input parameter was performed to obtain a linear input. The system was casted into an qLPV model, where the real input was computed through a scheduled inverse of the nonlinear input. This qLPV modeling approach was further developed and applied to an F-16[2,15] and to a Boeing 747-100/200.[18,20] The approach followed in this paper proposes a substitution of a so-called decomposition function $\mathcal{F}_{dec}$ by (scheduling parameter-dependent) functions linear in the scheduling vector. This decomposition function is formed by combining all of the terms of the nonlinear system that are not both, affine with respect to the nonscheduling states and control inputs, and function of the scheduling vector alone (after a coordinate change with respect to a single equilibrium point has been performed). The decomposition is carried out through a minimization procedure. The algorithm follows:

Starting from the nonlinear class of systems given in Eq. (4), choose an equilibrium point (z_{eq}, w_{eq}, u_{eq}), and perform the following change of coordinates:

$$\eta_z = z - z_{eq} \qquad \eta_w = w - w_{eq} \qquad \eta_u = u - u_{eq} \tag{10}$$

Subtract from Eq. (4) its own value at the chosen trim point and rearrange the terms of the resulting equation as follows (put together all of those terms that are not linear on the nonscheduling states and controls, and function of the scheduling vector):

$$\begin{bmatrix} \dot{\eta}_z - \dot{z}_{eq} \\ \dot{\eta}_w - \dot{w}_{eq} \end{bmatrix} = A_w(\eta_z + z_{eq})\eta_w + B(\eta_z + z_{eq})\eta_u$$

$$+ \mathcal{F}_{dec}(\eta_z, w_{eq}, u_{eq}) \tag{11}$$

The last term in the equation is given by

$$\mathcal{F}_{dec} = A_z(\eta_z + z_{eq})\eta_z + [A(\eta_z + z_{eq}) - A(z_{eq})] \begin{bmatrix} z_{eq} \\ w_{eq} \end{bmatrix}$$

$$+ [B(\eta_z + z_{eq}) - B(z_{eq})]u_{eq} + [k(\eta_z + z_{eq}) - k(z_{eq})] \tag{12}$$

The objective is to decompose $\mathcal{F}_{dec}(\eta_z, w_{eq}, u_{eq})$ into functions linear in $\eta_z \in \mathcal{R}^{n_z}$ and then substitute the result back into Eq. (11). The function decomposition is defined by

$$\mathcal{F}_{dec}(\eta_{eq}, w_{eq}, u_{eq}) = f_1(z)\eta_{z1} + f_2(z)\eta_{z2} + \cdots + f_n(z)\eta_{zn} \tag{13}$$

Note that this decomposition is exact if the desired functions $f_i(z)$ are selected as follows (see Ref. 7):

$$f_i(z) = \mathcal{F}_{dec}(z, z_{eq}, w_{eq}, u_{eq}) \left[\eta_{zi} \big/ \left(\eta_{z1}^2 + \eta_{z2}^2 + \cdots + \eta_{zn}^2 \right) \right] \tag{14}$$

These decomposition functions are required to be smooth and well-defined to avoid controller synthesis problems arising from discontinuities in the qLPV model.

It is necessary to select grid points for the scheduling parameters in order to make the problem finite dimensional (i.e., $z = [z_1(1 \ldots n) \; z_2(1 \ldots m) \; \cdots \; z_{n_z}(1 \ldots k)]^\top$). Once this is done, it is straightforward to obtain a particular solution $\tilde{f}_i(z) \in \mathcal{R}^{n \times m \times \cdots \times k}$ for each of the decomposition functions given by Eq. (14). It is necessary to set some additional constraints to ensure the smooth and well-defined conditions on the functions (i.e., bounds on their derivatives and coefficients). Also, because the only information available is the value of the decomposed function, Eq. (12), at the grid points, a numerical approximation approach is used to obtain $\tilde{f}_i(z)$ and their second derivatives. This could lead potentially to approximation errors that should be avoided or reduced by using smoothing techniques (see Chap. 3.3 in Ref. 21).

Two linear programs (LPs) are used to smooth the decomposition functions. The first LP minimizes the maximum absolute value of the second partial derivatives of the decomposition functions across all of the parameters in z. This first LP basically enforces a second-derivative continuity condition on the decomposition functions to ensure their smoothness. Mathematically it can be posed as follows:

$$\min\left(\max_{i,j} \max_z \left| \frac{\partial^2 f_i(z)}{\partial z_j^2} \right| \right) \tag{15}$$

subject to Eq. (13). The second linear program also minimizes the maximum absolute value but this time of the functions themselves, while constraining the optimization to a relaxation of the previous LP objective, that is, a threshold on the maximum second partial derivative of the functions. This objective is represented by the following minimization problem:

$$\left(\min_{i,j} \max_{i,j} \max_z |f_i(z)| \right)$$

subject to

$$\max_j \max_z \left| \frac{\partial^2 f_i(z)}{\partial z_j^2} \right| \le (1 + \delta_i)b_i \tag{16}$$

also subject to Eq. (13). The term δ_i is a design parameter used to relax the threshold on the constraint, and b_i is set to the maximum value of the second derivatives obtained from the first optimization across z and $f_i(z)$.

The LP algorithm that follows solves the decomposition problem. It is assumed that Eqs. (11) and (12) are available.

1) Find the particular solution to Eq. (14), $\tilde{f}_i(z)$, corresponding to the chosen grid vectors $z = [z_1(1 \ldots n) \; z_2(1 \ldots m) \; \cdots \; z_{n_z}(1 \ldots k)]^\top$.

2) Find the null space $\mathcal{N}_i(\eta_z)$ for the vector formed by combining the grid parameters of the scheduling states coordinate transformation $[\eta_{z_1} \; \eta_{z_2} \; \cdots \; \eta_{z_{n_z}}]^\top$. A singular value decomposition can be used to obtain the orthonormal basis that spans the null space.

3) Find the second partial derivatives for $\tilde{f}_i(z)$ and $\mathcal{N}_i(\eta_z)$ with respect to the scheduling parameters z. Because all of the information available is based on data points of the functions at the grid points, an approximate numerical differentiation technique is required. In this particular algorithm the Three Points Newton's Divided Difference Interpolation Formula (for example, see p. 60 in Ref. 21) is used.

4) Solve the initial linear program. It minimizes the second partial derivatives of the decomposition functions while allowing the value of the functions at the grid points to vary constrained by a user-defined bound $\Lambda \cdot \|\mathcal{F}_{\text{dec}}\|_\infty$.

$$\min_{X} \varepsilon \quad \text{subject to}$$

$$|X| \leq \Lambda \cdot \|\mathcal{F}_{\text{dec}}\|_\infty$$

$$\left| \frac{\partial^2}{\partial z^2} \cdot \mathcal{N}_i(\eta_z) \cdot X \right| \leq \left| \frac{\partial^2}{\partial z^2} \cdot \tilde{f}_i(z) \right| + \varepsilon \tag{17}$$

where Λ is a user-defined constant to limit the values of the decomposition functions and $\|\mathcal{F}_{\text{dec}}\|_\infty$ is the $\mathcal{H}_\infty$ norm of the decomposed function, Eq. (12). The initial guess for the LP is a zero vector of appropriate dimension. The solution X can then be used to calculate the new decomposition functions $f_i(z)_{\text{new}}$:

$$f_i(z)_{\text{new}} = \tilde{f}_i(z) + \mathcal{N}_i(\eta_z) \cdot X \tag{18}$$

It will also be used as an initial guess for the second LP and to calculate a new bound for the second derivatives.

5) Calculate the maximum second partial derivative for the new decomposition functions $f_i(z)_{\text{new}}$.

$$b_i = \max_{j} \max_{z} \left| \frac{\partial^2}{\partial z_j^2} \cdot f_i(z)_{\text{new}} \right| \tag{19}$$

This bound is used as a constraint on the derivatives in the following LP.

6) The second linear program minimizes the decomposition functions using as an initial guess the solution of the first LP and relaxing the maximum absolute value for the second derivatives by using the bound obtained in step 5.

$$\min_{Y} \varepsilon \quad \text{subject to}$$

$$|Y| \leq \infty$$

$$\left| \frac{\partial^2}{\partial z^2} \cdot \mathcal{N}_i(\eta_z) \cdot Y \right| \leq \left| \frac{\partial^2}{\partial z^2} \cdot \tilde{f}_i(z) \right| + (1 + \delta_i) \cdot b_i$$

$$|\mathcal{N}_i(\eta_z) \cdot Y| \leq |\tilde{f}_i(z)| + \varepsilon \tag{20}$$

where δ_i is a user-defined constant used to relax the gradient constraint. The final decomposition functions are then given by

$$f_i(z)_{\text{final}} = \tilde{f}_i(z) + \mathcal{N}_i(\eta_z) \cdot Y \tag{21}$$

Substituting in Eq. (11) the decomposition function $\mathcal{F}_{\text{dec}}$ by its right-hand side equivalent of Eq. (13) [note that it now contains the final decomposition functions from Eq. (21)] and noting the earlier coordinate change ($\eta_z + z_{\text{eq}} = z$), the function substitution qLPV model is obtained:

$$\begin{bmatrix} \dot{z} \\ \dot{w} \end{bmatrix} = [A_z(z) + f_i(z)_{\text{final}} \quad A_w(z)] \cdot \begin{bmatrix} z - z_{\text{eq}} \\ w - w_{\text{eq}} \end{bmatrix} + B(z) \cdot [u - u_{\text{eq}}] \tag{22}$$

where the selected trim condition ($z_{\text{eq}}, w_{\text{eq}}, u_{\text{eq}}$) is fixed.

Disadvantages of this approach include several open questions, among them the importance of the trim point chosen and its effects in the subsequent LPV controller synthesis stage. Also, the typical problem of selecting the grid vector, which is almost endemic to all of the LPV techniques, is present here. The advantage is that the dependency of the approach does not rely on the feasible trim region but on a user-defined grid.

III.   Boeing 747-100/200

The aircraft model used in this paper is a Boeing 747 series 100/200. This aircraft was chosen because its wide array of characteristics (leading- and trailing- edge flaps, spoilers, variety of control surfaces, four fan jet engines, etc.) make of it the perfect representative for any of the commercial airplanes flying today, and thus an ideal test bed to prove the versatility of the LPV modeling and design techniques. The Boeing 747 is an intercontinental wide-body transport with four fan jet engines designed to operate from international airports.

The body-axes longitudinal motion of the Boeing 747, not including flexible effects, can be described by the following differential equations (assuming no wind):

$$\dot{\alpha} = \frac{[-F_x \cdot s_\alpha + F_z \cdot c_\alpha]}{m \cdot V_{TAS}} + q \tag{23}$$

$$\dot{q} = c_7 \cdot M_y \tag{24}$$

$$\dot{\theta} = q \tag{25}$$

$$\dot{V}_{\text{TAS}} = \frac{1}{m} \cdot [F_x \cdot c_\alpha + F_z \cdot s_\alpha] \tag{26}$$

$$\dot{h}_e = V_{\text{TAS}} \cdot c_\alpha \cdot s_\theta - V_{\text{TAS}} \cdot s_\alpha \cdot c_\theta \tag{27}$$

The states of the system are angle of attack α, pitch rate q, pitch angle θ, true airspeed V_{tas}, and altitude h_e. Longitudinal control is performed through a movable horizontal stabilizer σ, four elevator segments, and the thrust from the four engines Tn_i. Pitch trim is provided by the horizontal stabilizer, and under normal operation the inboard and outboard elevators move together, $\delta_E = \delta_{E_I} = \delta_{E_O}$.

The body-axes aerodynamic forces and moments are given by

$$F_x = -\bar{q} \cdot S \cdot [C_D \cdot c_\alpha - C_L \cdot s_\alpha] + \sum_{i=1,4} Tn_i - m \cdot g \cdot s_\theta \tag{28}$$

$$F_z = -\bar{q} \cdot S \cdot [C_D \cdot s_\alpha + C_L \cdot c_\alpha] - 0.0436 \cdot \sum_{i=1,4} Tn_i + m \cdot g \cdot c_\theta \tag{29}$$

$$M_y = \bar{q} \cdot S \cdot \bar{c} \cdot \left\{ C_m - \frac{1}{\bar{c}}[(C_D \cdot s_\alpha + C_L \cdot c_\alpha) \cdot \bar{x}_{\text{cg}} - (C_D \cdot c_\alpha - C_L \cdot s_\alpha) \cdot \bar{z}_{\text{cg}}] + \frac{\bar{c} \cdot \dot{\alpha}}{V_{\text{TAS}}} C_{m_{\dot{\alpha}}} \right\}$$

$$+ \bar{q} \cdot S \cdot \bar{c} \cdot \left\{ -\frac{\bar{x}_{\text{cg}}}{\bar{c}} \cdot C_{L_\alpha} \cdot c_\alpha \right\} + \sum_{i=1,4} Tn_i \cdot zeng_i \tag{30}$$

The aerodynamic data for the Boeing 747-100/200 were obtained from Refs. 22 and 23. Because the full set of aerodynamic coefficients is deemed too complex for modeling and control design, a low-fidelity model of the aircraft is obtained through a two-step simplification study. First, an analytical study of the importance of each stability derivative with respect to the nominal value of the total aerodynamic coefficient is performed. Second, open-loop time simulations are performed to refine the low-fidelity model and to ascertain the validity of the final reduced set. In Fig. 1, time responses for both aerodynamic systems, the high-fidelity (Hi-Fi) and the final low-fidelity models (Lo-Fi), trimmed at the design point 1 given in Table 1 are shown. The command is a 1.2-deg step input of the elevator applied at time = 15 s. The difference in the angle of attack is caused by software constraints, that is, the trim subroutine used the angle of attack as an independent variable. It is noted that the similitude for the short period mode (the most critical) and small differences in the period and amplitude for the phugoid. The details of this reduction can be obtained in Ref. 18, where the six aerodynamic coefficients were studied. This final low-fidelity model (Lo-Fi) is used in the LPV modeling stage. Note that the final goal of

any controller is to control the actual high-fidelity nonlinear model, but in order to use LPV control techniques an LPV model is required. These LPV models must capture the main characteristics of the nonlinear system but yet be sufficiently simple to handle.

The reduced aerodynamic coefficients for the longitudinal motion are given next ($\tilde{x}_{c.g.} = 1.45 - 1.8x_{c.g.}$):

$$C_L = C_{L_{basic}}(\alpha_w, M) + \frac{dC_L}{dq}(he, M) \cdot \frac{q \cdot \bar{c}}{2 \cdot V_{TAS}} \cdot \tilde{x}_{c.g.}$$

$$+ K_\alpha(\alpha_w) \cdot \left[\frac{dC_L}{d\delta_{E_I}}(he, M) + \frac{dC_L}{d\delta_{E_O}}(he, M) \right] \cdot \delta_E \quad (31)$$

$$C_D = C_{D_{Mach}}(M, C_L^*) \quad (32)$$

$$C_m = C_{m_{basic}}(\alpha_w, M) + \frac{dC_{m0.25}}{dq}(he, M) \cdot \frac{q \cdot \bar{c}}{2 \cdot V_{TAS}}$$

$$+ K_\alpha(\alpha_w) \cdot \frac{dC_{m0.25}}{d\sigma}(he, M) \cdot \sigma_{F.R.L.}$$

$$+ K_\alpha(\alpha_w) \cdot \left[\sum_{i=I,O} \frac{dC_{m0.25}}{d\delta_{E_i}}(he, M) \right] \cdot \delta_E \quad (33)$$

The lift coefficient C_L depends only on the effects of the pitch rate, the elevators (which enter linearly), and the basic component $C_{L_{basic}}$. The pitching-moment aerodynamic coefficient C_m has the same dependencies as the lift coefficient plus a term to account for the effect of the horizontal stabilizer, which also enters linearly. The drag coefficient C_D has only one term accounting for the effect of the Mach number $C_{D_{Mach}}(M, C_L^*)$. This stability derivative depends on
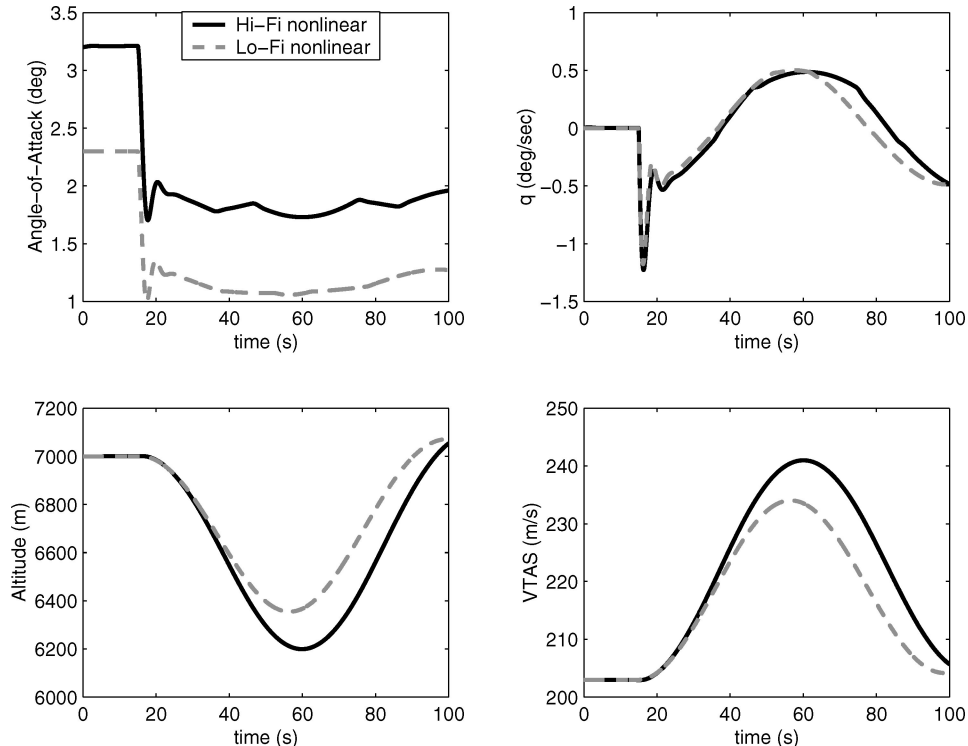
Mach number and the first two terms in the lift coefficient equation (note the dependency on the pitch rate state q):

$$C_L^* = C_{L_{basic}} + \frac{dC_L}{dq} \cdot \frac{q \cdot \bar{c}}{2 \cdot V_{TAS}} \cdot [1.45 - 1.8x_{c.g.}] \quad (34)$$

IV. Software

The software used in this project to simulate and analyze the behavior of the Boeing 747 is an enhanced version of Flight Lab 747, FTLAB747, and its predecessor, Delft University Aircraft Simulation and Analysis Tool, DASMAT. These programs were developed by Delft University of Technology, The Netherlands (see Refs. 24 and 25) as a learning tool to simulate and analyze aircraft dynamics, for example, of a twin-engined business jet, the Cessna Citation 500. FTLAB747 was developed specifically to study the EL AL Israel Airlines (ELY) 1862 crash accident on 4 October 1992 near Amsterdam and is a particularization of DASMAT registering changes pertaining the Boeing 747 and accounting for flight failures. The version used in this paper, FTLAB747v61/65, has been enhanced to use the Boeing 747-100/200 as a benchmark aircraft for fault tolerant and fault detection research (see Ref. 26).

The program is based in MATLAB®/Simulink and offers a wide array of simulation and analysis tools (trim, linearization, simulation, and flight visualization). The structure of the program is modular and exploits the advantage of decoupling general from aircraft specific dynamics, allowing easy implementation of other aircraft. The trim subroutine returns the control inputs and the aircraft and engine states corresponding to a flight condition where the linear and angular accelerations are zero. It provides six different options for trimming the aircraft: straight-and-level, push-over/pull-over, level turn, thrust-stabilized turn, beta trim, and specific power turn (see Refs. 24 and 27). Each condition requires different constraints and independent variables that are primarily the main flight controls and the aircraft/engine states. Because of the specific approach used to trim the aircraft, there are instances when differences on the states or control inputs are found between the high- and low-fidelity models.

V. Quasi-LPV Models

In this section the details pertaining the qLPV modeling for the Boeing 747 aircraft are presented. A comparison and discussion

Table 1 Design and trim points

Condition	Altitude, m	Velocity, m/s	α, deg
Design point 1	7,000	203	2.299
Design point 2	10,000	195	3.661
Trim point 1	8,500	175	5.742
Trim point 2	8,500	223	1.895

Fig. 1 Open-loop time response for the high-fidelity and low-fidelity aerodynamic models.

of the three modeling techniques is postponed for the following section.

Generally, the scheduling variables for an aircraft are a combination of altitude he, velocity V_{tas}, angle of attack α, and/or variables that register changes in those parameters such as Mach number M or dynamic pressure $\bar{q}$. In the present case to avoid excessive approximations and because of the dependencies of the stability derivatives, the scheduling variables selected are angle of attack α, true airspeed V_{TAS}, and altitude he. Hence, the corresponding system vectors in Eq. (4) are $z = [\alpha \ \ V_{TAS} \ \ h_e]^T$, $w = [q \ \ \theta]^T$, and $u = [\delta_e \ \ \sigma \ \ Tn]^T$. Because the scheduling variables are all states of the system, this results in a quasi-LPV model.

Rewriting the Boeing 747 longitudinal nonlinear equations of motion as in Eq. (4), some of the nonscheduling states enter the system in a nonlinear fashion. Specifically, the pitch angle enters the A and K matrices through trigonometric functions (s_θ and c_θ) and the pitch rate the K matrix through the drag coefficient C_D.

To transform the nonlinear entries, a linearization with respect to a trim value is performed for the pitch angle θ, where ∇_θ is the difference between the state and a trim point:

$$\cos\theta = \cos\theta_{eq} - \sin\theta_{eq} \cdot \nabla_\theta \tag{35}$$

$$\sin\theta = \sin\theta_{eq} + \cos\theta_{eq} \cdot \nabla_\theta \tag{36}$$

A similar linearization for the Mach-number effect on drag stability derivative $C_{D_{Mach}}$ is performed (see Ref. 18 for more details):

$$C_{D_{Mach}}\left(M, C_L^*\right) \equiv C_{D_{Mach}}(\alpha_w, q, M, h_e) \tag{37}$$

$$C_{D_{Mach}}(\alpha_w, q, M, h_e) \approx C_{D_{Mach}}(\alpha_w, q_{eq}, M, h_e)$$

$$+ \left.\frac{\partial C_{D_{Mach}}}{\partial q}\right|_{eq} \cdot (q - q_{eq}) \tag{38}$$

After these approximations the new dependencies of the equations of motion are as required[18]:

$$\dot{x}_{long} = A(\rho, \theta_{eq}) \cdot x_{long} + B(\rho) \cdot y_{long} + k(\rho, \theta_{eq}, q_{eq}) \tag{39}$$

where the scheduling vector is $\rho = (\alpha, V_{TAS}, he)$, the longitudinal states $x_{long} = [\alpha \ \ q \ \ V_{TAS} \ \ \nabla_\theta \ \ he]^\top$, and the longitudinal control inputs $y_{long} = [\delta_e \ \ \sigma \ \ Tn]^\top$.

Next, it is necessary to study the feasible trim map since the state transformation and the Jacobian linearization are dependent on the equilibrium set. For flight control, it is common to trim by zeroing out the aerodynamics forces and moments ($28 \rightarrow 30$). It is easy to verify that $q_{eq} = 0$ and $\theta_{eq} = \alpha$ (straight-level flight). Figure 2 shows the acceptable trim region in terms of the angle of attack and the true airspeed at a given altitude, in this case 7000 m.
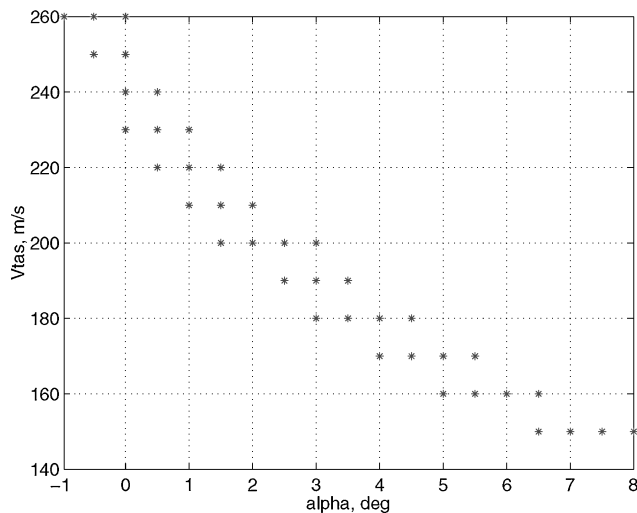

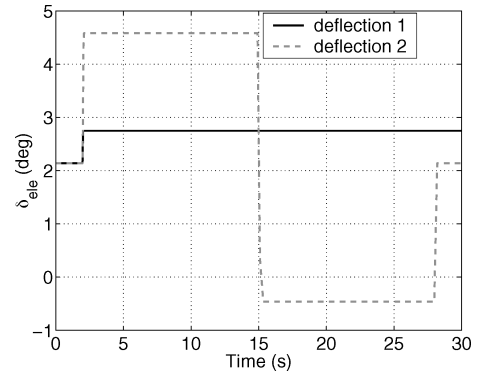**Fig. 2   Trim map at 7000 m.**

Fig. 3   Control inputs for nonlinear and qLPV simulations.

For the simulation of the high-fidelity nonlinear and qLPV models, two maneuvers are applied (see Fig. 3). The first command input, deflection 1, is a small perturbation of the pilot's column equal to 0.5 deg step starting after 2 s and in effect until the end of the simulation. The second command, deflection 2, is a deflection of the column composed by a ± 2 deg doublet applied from time = 2 s until time = 28 s. This second command results in a maneuver that takes the aircraft sufficiently away from the trim region to compare the qLPV models behaviour outside that region. The deflection of the column control by the pilot is equivalent, in this case, to elevator deflections of 0.61 and 4.45 deg for each of the deflections.

A.   Jacobian Linearization Quasi-LPV Model

The Jacobian quasi-LPV model is quite straightforward to obtain from Eq. (39). The typical approach performs linearizations with respect to a set of trim points to obtain a family of LTI plants that form the qLPV model. In the present case, it is possible to obtain a general state-space description of the model where all of the dependencies are in terms of the scheduling variables by performing a linearization with respect to a generic trim point. Using the trim subroutine mentioned before, a trim map in terms of the scheduling vector can be found (although there is no assurance that feasible trim values exist for a particular flight condition). This general state-space realization together with the associated trim map compose the Jacobian qLPV model. In Ref. 18 the quasi-LPV model is given in detail together with the complete derivation for one of the states.

This general description approach is possible because the preceding generic linearization can be written in an analytical, well-defined form, and only interpolation of the trim map is necessary. Most commonly, only the LTI state-space matrices can be found, and interpolation among members of the family of LTI plants is necessary. A drawback of this analytical approach, besides the tediousness of deriving the analytical equations, is the requirement to find partial derivatives for the aerodynamic coefficients look-up tables. This is a task prone to errors because usually it requires extrapolation of the available aerodynamic data and numerical differentiation. For this particular application, the trim map used required quite a dense grid to avoid numerical extrapolation problems at the boundary of the trim region. This grid density resulted in a more accurate model within the trim region albeit with a greater computational demand.

B.   State Transformation Quasi-LPV Model

For the qLPV modeling of the Boeing 747-100/200 longitudinal motion, the starting point is Eq. (39), equivalent to the class of nonlinear systems given by Eq. (4). The equilibrium functions and their derivatives are quite easy to obtain because $q_{eq}(\rho) = 0$ and $\theta_{eq}(\rho) = \alpha$, and for any value of $\rho = (\alpha, V_{TAS}, h_e)$ their derivatives are

$$\begin{bmatrix} \dfrac{\partial q_{eq}(\rho)}{\partial \rho} \\[2mm] \dfrac{\partial \theta_{eq}(\rho)}{\partial \rho} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \tag{40}$$

Therefore, the qLPV system for the longitudinal motion of the Boeing 747 using the state transformation approach is a basic reorganization of the state-space coefficients in the nonlinear equation (39). The trim map used for the Jacobian qLPV model is also used in this approach. The map is required because it is not possible to calculate an analytical equation for the equilibrium functions, that is, the trim map is a de facto numerical representation of the equilibrium functions $w_{eq}(\rho)$ and $u_{eq}(\rho)$ for a given ρ. Hence, this approach has the same shortcomings and advantages as the previous modeling technique in terms of grid density.

C. Function Substitution Quasi-LPV Model

The same nonlinear equation for the Boeing 747-100/100 used in the other two approaches, Eq. (39), is also the departure point for this method. This approach models the nonlinear system around a unique trim point, and, as it was seen before, there are two successive minimizations: one for the second partial derivatives of the desired decomposition functions and the other for the decomposition functions. There are four main designer knobs that can affect the resulting qLPV model: the trim point, the grid of the particular solution (step 1 in the function substitution algorithm), the first LP bound Λ (step 4), and the second LP bound δ_i (step 6).

Before applying the algorithm, the requirement is to rewrite the nonlinear system as in Eqs. (11) and (12), the latter is the function to be decomposed $\mathcal{F}_{dec}$. This is done using Eq. (39) and the selected trim point. After selecting the grid and the bounds, application of the algorithm yields the function substitution qLPV model in the form of Eq. (22).

The selection of the trim point for the design of the function qLPV model seems to be not that critical (at least for this case and looking at the open-loop time responses). Models are obtained at several trim points in the flight envelope although only two are used in this paper (see the design points in Table 1). To appropriately compare them, the resulting qLPV models are trimmed at different points in the flight envelope (see the trim points in the same table). Figure 4 shows the time responses for deflection 2 for the nonlinear models and the two function qLPV models' all trimmed at trim point 1. It is observed that both qLPV models follow quite closely the nonlinear models although the model designed at the design

point 2 is slightly better. This is mostly because of its closeness in speed to the trim point selected. Figure 5 show the time responses for the same deflection and models but trimmed at trim point 2. This time the qLPV models' responses seem to be slightly off with respect to the high-fidelity model for the phugoid mode. But when compared to the low-fidelity (which was the baseline model for the qLPV modeling stage), it is noticed that this is a shortcoming of this nonlinear model rather than of the qLPV models. Indeed, the qLPV model designed closer in speed to the trim point, design point 1, is closer this time to the low-fidelity nonlinear response. The final selected design point for the qLPV model for the sequel is design point 1.

The choice of grid in the first step of the qLPV modeling algorithm is found to be more significant. Table 2 shows three of the grids used to assess this influence. It is natural to assume that a finer grid will yield better results. This seems to be actually the case although some of the scheduling parameters have more impact than others and there is interplay among them based on the modes they excite. Specifically, the angle of attack (short period) and airspeed (phugoid) are the predominant parameters, with improvements in one mode adversely affecting the other. Using a very fine grid yielded models of a more oscillatory nature. It is suspected that the computational demand of these refined grids might have introduced undesirable errors. Figure 6 shows a comparison with respect to the high-fidelity nonlinear model of the qLPV obtained at the selected design point, design point 1, and using the grids given in Table 2. Observe how the model for grid number two is closer for the angle of attack (improved short period) but further away for the altitude and speed (phugoid). The grid number 3 is selected because it shows the best overall fit to the nonlinear model.

The user-defined bounds Λ and δ_i affect, in a proportional manner, the maximum value of the coefficients in the first LP and the

Table 2   Scheduling grids, a:b:c => from a to c every b

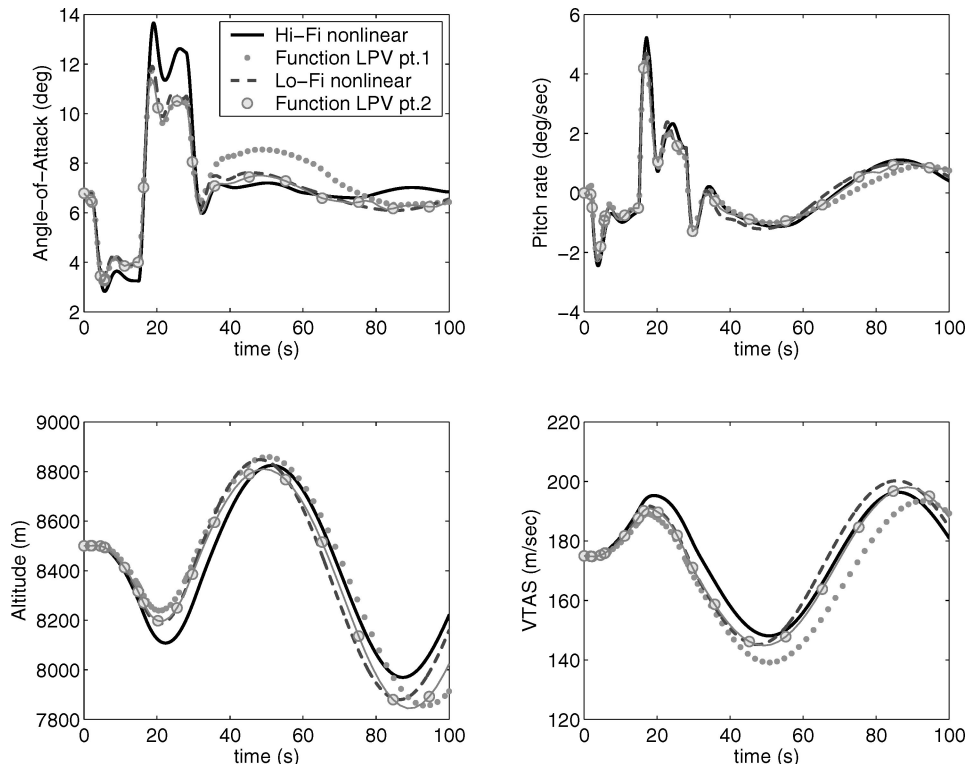| Grid	Altitude, m	Velocity, m/s	α, deg
1	[800:600:11,000]	[80:10:300]	[−2:3:30]
2	[0:1000:11,000]	[80:20:300]	[−2:5:30]
3	[5000:500:11,000]	[80:10:300]	[−2:5:30]

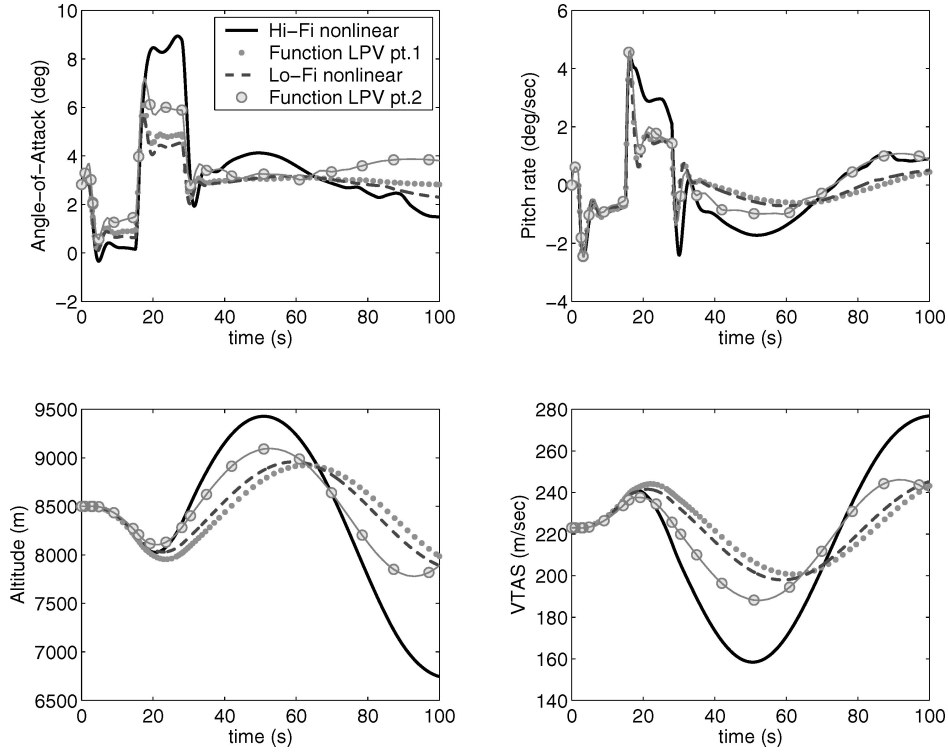**Fig. 4   Function qLPV models: design point variations (trim point 1, grid 1, deflection 2).**

Fig. 5 Function qLPV models: design point variations (trim point 2, grid 1, deflection 2).
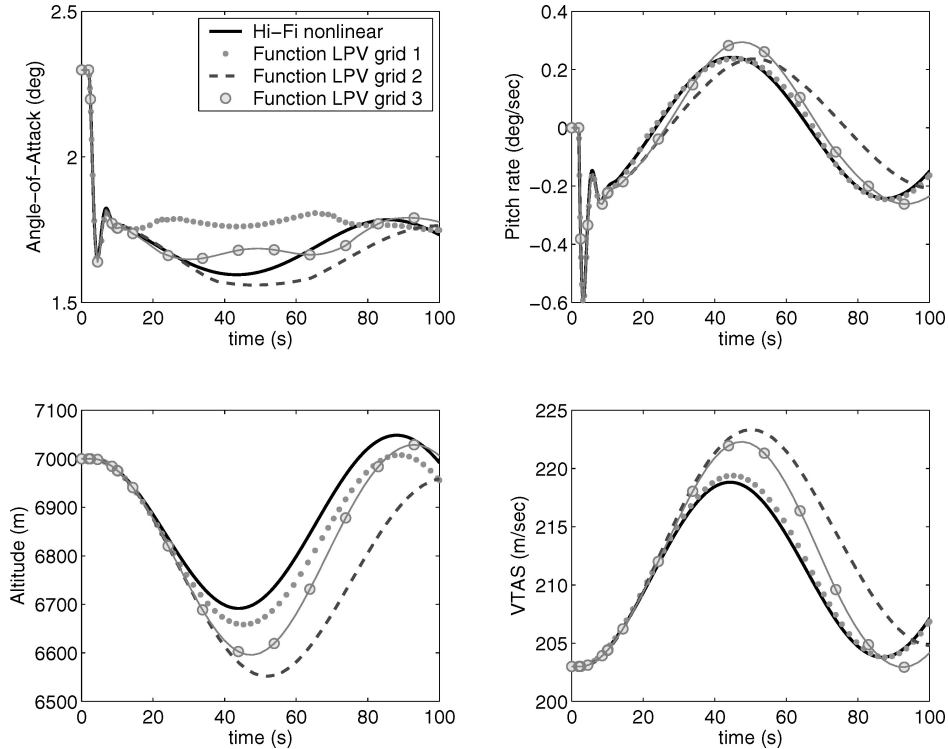

Fig. 6 Function qLPV models: grid variations (design point 1, deflection 1).

maximum value of the second derivatives in the second LP, respectively. It is observed that unless both bounds are selected to be of great dimension the succession of the two linear programs tend to downplay their effects to the point of making them negligible.

VI.   Comparison Study

This section provides a comparative study of the three qLPV modeling approaches. Open-loop time responses are used to compare the qLPV models and the high-fidelity (Hi-Fi) nonlinear model.

A comparison between the qLPV models is also drawn based on the difficulty of the modeling technique and the qLPV models advantages and shortcomings.

Figure 7 presents the open-loop responses to deflection 1 of the four models: Jacobian qLPV (dotted), state transformation qLPV (dashed), function substitution qLPV (solid with circle marker), and the Hi-Fi nonlinear model (thick solid). All of the qLPV models are obtained at design point 1. The Jacobian and state transformation qLPV models are indistinguishable because of the use
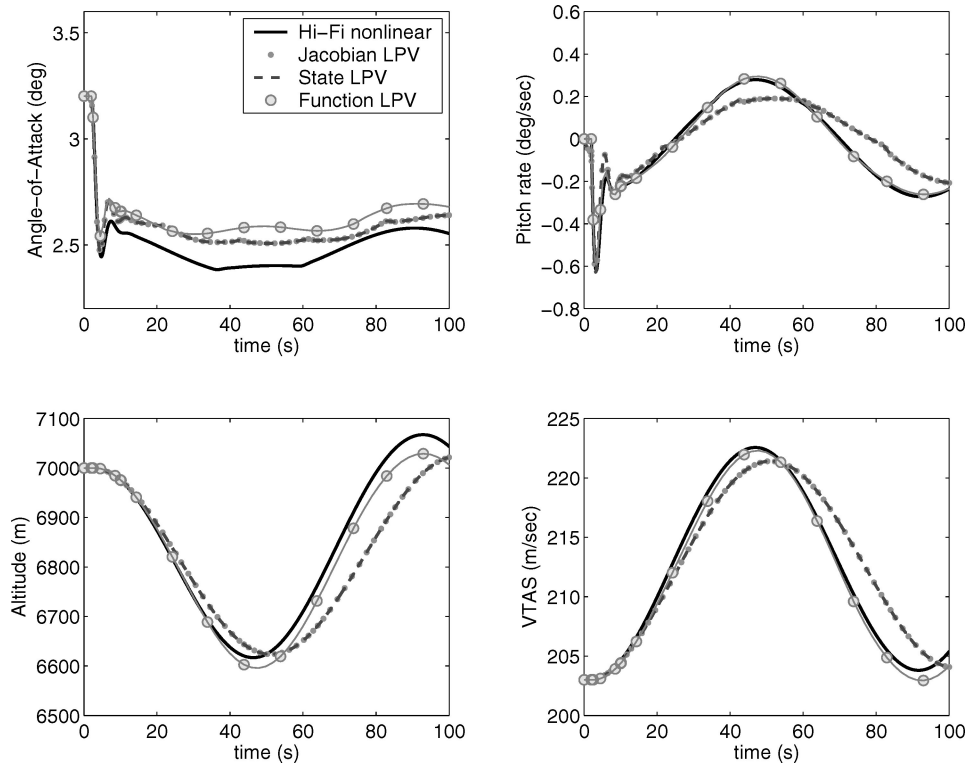
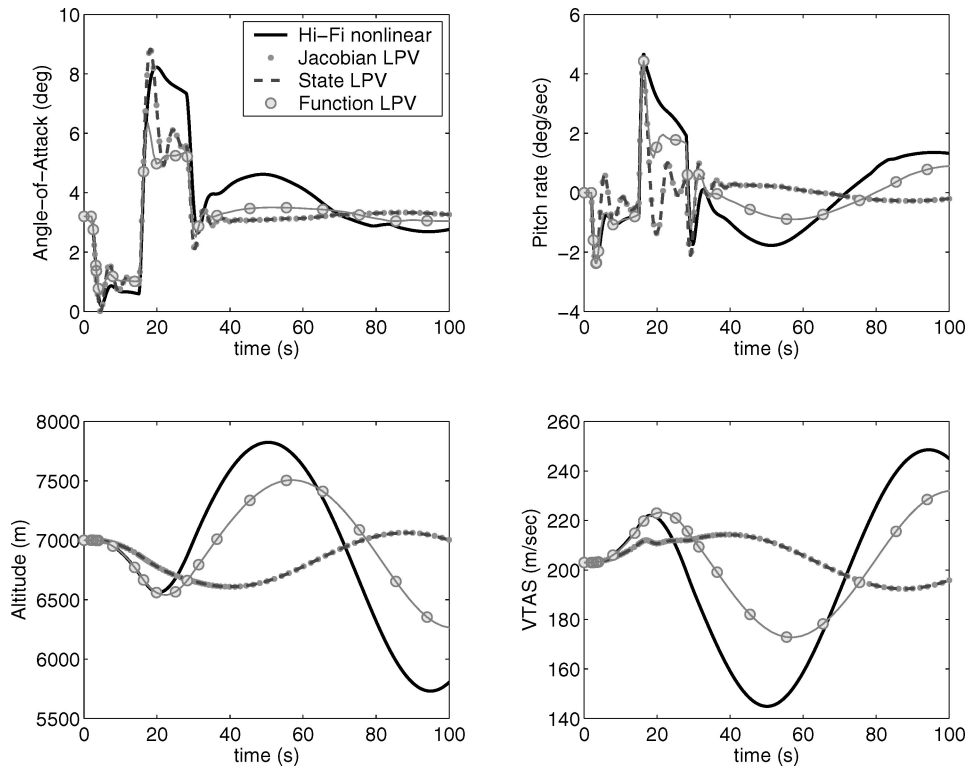Fig. 7   qLPV and nonlinear models (design point 1, deflection 1).

Fig. 8   qLPV and nonlinear models (design point 1, deflection 2).

of the same dense trim map. With respect to the nonlinear model, they are closer in magnitude for the angle of attack although the function qLPV model is able to capture better the dynamic behavior for that state. For the rest of the states, it is easily seen that the function qLPV model almost perfectly matches the nonlinear model behavior.

In Fig. 8 the time responses for deflection 2 are given. This maneuver results in a stronger model behavior that takes the aircraft

outside its trim region. Again, the Jacobian and state-transformation qLPV models are indistinguishable although it is clear that they do not provide a good response (neither in magnitude nor periodicity) compared to the nonlinear model modes. The function qLPV model does not seems as close to the Hi-Fi nonlinear model as in the preceding simulation, but it is able to capture the dynamic behavior much better than the other two models. Indeed, the offset in the function qLPV is not so much a shortcoming of the modeling

procedure as a limitation of the Lo-Fi nonlinear model used for the qLPV stage (see preceding section).

In terms of the difficulty of obtaining the quasi-LPV models, it is noted that once the nonlinear equations are transformed into the class of nonlinear systems characterized by Eq. (4) all of the quasi-LPV models are relatively easy to obtain. The Jacobian approach presented an additional difficulty because of the approach taken, which involved representing the qLPV model through a general analytic equation and a trim map. This approach required obtaining a number of partial derivatives for the aerodynamic coefficients look-up tables, which resulted in a simple but tedious task. This can be avoided by simply using one of the many available linearization routines to find a family of LTI plants. The state transformation, for the Boeing 747 case, was almost a simple reshuffling of the terms in Eq. (4). The function substitution involved an optimization routine aimed at decomposing the nonlinear function given by Eq. (12). This optimization is for the major part a systematic routine except for the initialization stage that needs to be adapted for the particular aircraft considered (an easy, straightforward process).

Summarizing the advantages of each model, it is observed that the Jacobian linearization approach is the most widespread methodology, and it has a proven theoretical base. Indeed, the typical Jacobian LPV modeling approach is the easiest of the three techniques presented. Transfer of knowledge and technology is also natural for this approach. It is also applicable to the widest class of nonlinear systems because it is valid for any nonlinear system that can be linearized at its equilibrium points. The state transformation provides an exact LPV model of the nonlinear system, which can be critical for systems where approximations are not feasible. The function substitution is not constrained by the feasible trim region, and in overall terms (for this present case at least) it better characterizes the behavior of the nonlinear model. All of the approaches allow the use of LPV control synthesis techniques.

With respect to the disadvantages and limitations of the different modeling approaches, the Jacobian LPV approach is a first-order approximation of the system to equilibrium points, and generally it is not possible to capture the transient behavior of the nonlinear model. Similarly, the state-transformation model depends on the existence of continuously differentiable trim functions for the nonscheduling states and inputs $w_{eq}(\rho)$ and $u_{eq}(\rho)$. Unfortunately there are no assurances of their existence. Both approaches, the Jacobian and state transformation, are limited in their usefulness (the fidelity to the nonlinear model) by the range of the feasible trim map. If this region is small, as it is in the present case, the simulation can be hampered by the limited range in which the qLPV model is meaningful. Also, in the case the LPV model is obtained by forming a family of LTI plants evaluated at selected trim points, the required interpolations to model the behavior of the system in between trim points can result in approximation errors. This is true for both Jacobian and state LPV models. The function substitution method still suffers from several open questions, for example, the influence that the choice of the design point has on the LPV model and LPV control synthesis, the correct characterization of the smoothness of the decomposition functions and the effect on the qLPV model, and numerical effects of the grid and their influence.

VII. Conclusion

In this paper three quasi-linear-parameter-varying (LPV) modeling techniques have been studied and their application to the Boeing 747 longitudinal motion presented. Each approach differs in the nonlinear class of models they cover, theoretical knowledge, easiness of model design, and trim restriction mainly.

The Jacobian linearization and the state transformation approaches to LPV modeling are better known and the necessary theoretical developments more straightforward, but they are the most restrictive in terms of operational envelope because they require the existence of trim functions or points. The theory for the function substitution still has several open questions to be addressed, but this approach provides the widest envelope for the model. The easiness of model design is relatively similar for all of them; probably the typical Jacobian approach is the easiest, although this is problem

dependent and hence is difficult to measure. The function substitution qLPV seems to be the more accurate model for the Boeing 747-100/200 aircraft.

The final goal of developing LPV/qLPV models is to enable the use of LPV control techniques, which will be applied at the end to the high-fidelity nonlinear system. Because the true nonlinear system operates at trim and nontrim points in the operational envelope, it is noted that qLPV models that capture the behavior of the nonlinear model for the widest envelope have a greater chance of providing with the necessary tools for the control designer. Also, there are no theoretical results quantifying which LPV model is optimal for control design. This is an inherently difficult problem because the LPV model is meant to be an approximation of the nonlinear system. It is unclear how to select the "best" LPV model, which results in the best LPV controller for the nonlinear system.

References

[1]Balas, G. J., Fiahlo, I., Packard, A., Renfrow, J., and Mullaney, C., "On the Design of the LPV Controllers for the F-14 Aircraft Lateral-Directional Axis During Powered Approach," *American Control Conference*, American Automatic Control Council, Evanston, IL, Vol. 1, 1997, pp. 123–127.

[2]Shin, J. Y., "Worst-Case Analysis and Linear Parameter-Varying Gain-Scheduled Control of Aerospace Systems," Ph.D. Dissertation, Dept. of Aerospace and Engineering Mechanics, Univ. of Minnesota, Minneapolis, Oct. 2000.

[3]Balas, G. J., Mueller, J., and Barker, J., "Application of Gain-Scheduled Multivariable Control Techniques to the F/A-18 Systems Research Aircraft," AIAA Paper 1999-4206, Aug. 1999; also in *Proceedings of the AIAA Applied Aerodynamics Conference*, AIAA, Reston, VA, 1999.

[4]Papageorgiou, G., and Glover, K., "Design, Analysis and Flight Testing of a Robust Gain Scheduling Controller for the VAAC Harrier," DERA, Tech. Rep. TR-CUED/F-INFENG/TR.368, Univ. of Cambridge, U.K., Feb. 2000.

[5]Balas, G. J., Ryan, J., Shin, J., and Garrard, W., "A New Technique for Design of Controllers for Turbofan Engines," AIAA Paper 1998-3751, July 1998; also in *Proceedings of the AIAA/ASME Joint Propulsion Conference*, AIAA, Reston, VA, 1998.

[6]Wolodkin, G., Balas, G. J., and Garrard, W., "Application to Parameter Dependent Robust Control Synthesis to Turbofan Engines," AIAA Paper 98-0973, Jan. 1998; also in *Proceedings of the AIAA Aerospace Science Meeting*, AIAA, Reston, VA, 1998.

[7]Tan, W., "Applications of Linear Parameter-Varying Control Theory," Master's Thesis, Dept. of Mechanical Engineering, Univ. of California at Berkeley, May 1997.

[8]Wu, F., Packard, A., and Balas, G. J., "LPV Control Design for Pitch-Axis Missile Autopilots," *34th Conference on Decision and Control*, Vol. 1, IEEE Press, New York, 1995, pp. 188–193.

[9]Bennani, S., Sluis, R., Schram, G., and Mulder, J. A., "Control Law Reconfiguration Using Robust Linear Parameter Varying Control," AIAA Paper 1999-4137, Aug. 1999; also in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 1999.

[10]Ganguli, S., Marcos, A., and Balas, G. J., "Reconfigurable LPV Control Design for Boeing 747-100/200 Longitudinal Axis," *American Control Conference*, American Automatic Control Council, Evanston, IL, Vol. 5, 2002, pp. 3612–3617.

[11]Shamma, J., and Cloutier, J., "Gain-Scheduled Missile Autopilot Design Using Linear Parameter Varying Transformations," *Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 2, 1993, pp. 256–261.

[12]Papageorgiou, G., "Robust Control System Design $\mathcal{H}_\infty$ Loop Shaping and Aerospace Application," Ph.D. Dissertation, Dept. of Engineering, Univ. of Cambridge, Cambridge, England, U.K., July 1998.

[13]Leith, D. J., and Leithead, W. E., "Survey of Gain-Scheduling Analysis and Design," *International Journal of Control*, Vol. 73, No. 11, 2000, pp. 1001–1025.

[14]Rugh, W. J., and Shamma, F., "Research on Gain Scheduling," *Automatica*, Vol. 36, 2000, pp. 1401–1425.

[15]Bhattacharya, R., Balas, G. J., Kaya, M. A., and Packard, A., "Nonlinear Receding Horizon Control of an F-16 Aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 5, 2002, pp. 924–931.

[16]Leith, D. J., and Leithead, W. E., "Comments on the Prevalence of Linear Parameter-Varying Systems," Dept. Electronic and Electrical Engineering, Tech. Rept., Univ. of Strathclyde, Glasgow, Scotland, 1999.

[17]Becker, G., "Quadratic Stability and Performance of Linear Parameter Dependent Systems," Ph.D. Dissertation, Dept. of Engineering, Univ. of California, Berkeley, Dec. 1993.

[18]Marcos, A., "A Linear Parameter Varying Model of the Boeing 747-100/200 Longitudinal Motion," Master's Thesis, Dept. of Aerospace and Engineering Mechanics, Univ. of Minnesota, Minneapolis, Jan. 2001.

[19]Szászi, I., Marcos, A., Balas, G. J., and Bokor, J., "LPV Detection Filter Design for Boeing 747-100/200," AIAA Paper 2002-4347, Aug. 2002; also in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 2002.

[20]Marcos, A., and Balas, G. J., "Linear Parameter Varying Modeling of the Boeing 747-100/200 Longitudinal Motion," AIAA Paper 2001-4347, Aug. 2001; also in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 2002.

[21]Hildebrand, F. B., *Introduction to Numerical Analysis*, 2nd ed., McGraw–Hill, New York, 1974, Chap. 3.3.

[22]Hanke, C., "The Simulation of a Large Jet Transport Aircraft. Volume I: Mathematical Model," NASA CR-1756, Tech. Rept., March 1971.

[23]Hanke, C., and Nordwall, D., "The Simulation of a Jumbo Jet Transport Aircraft. Volume II: Modeling Data," NASA CR-114494/D6-30643-VOL-2, Tech. Rept., Sept. 1970.

[24]van der Linden, C. A. A. M., "DASMAT: Delft University Aircraft Simulation Model and Analysis tool," Technical Univ., Tech. Rept. LR-781, Delft, The Netherlands, Sept. 1996.

[25]Smaili, M. H., "FLIGHTLAB 747: Benchmark for Advanced Flight Control Engineering," Technical Univ., Tech. Rept., Delft, The Netherlands, Feb. 1999.

[26]Marcos, A., and Balas, G. J., "A Boeing 747-100/200 Aircraft Fault Tolerant and Fault Diagnostic Benchmark," Dept. of Aerospace Engineering and Mechanics, Univ. of Minnesota, Tech. Rept. AEM-UoM-2003-1, Minneapolis, June 2003.

[27]Stevens, B., and Lewis, F., *Aircraft Control and Simulation*, 2nd ed., Wiley, New York, 1992, Chap. 3.