JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 31, No. 4, July–August 2008

Engineering Notes

ENGINEERING NOTES are short manuscripts describing new developments or important results of a preliminary nature. These Notes should not exceed 2500 words (where a figure or table counts as 200 words). Following informal review by the Editors, they may be published within a few months of the date of receipt. Style requirements are the same as for regular contributions (see inside back cover).

Stability in Ornithopter Longitudinal Flight Dynamics

John M. Dietl* and Ephrahim Garcia†
Cornell University, Ithaca, New York 14853

DOI: 10.2514/1.33561

I. Introduction

PRACTICAL ornithopter designs have long eluded the aerospace engineering profession as engineers have systematically improved performance in other aerodynamic regimes. The design of flapping-wing vehicles has progressed, however, leaving a rich history of improvement and imagination. Leonardo da Vinci provided an early impetus to powered flight, misleading engineers for centuries from the simpler fixed-wing design that has proven practical [1]. Alphonse Penaud's rubber-band powered model showed the possibility of powered ornithopter construction [2]. After the Wright brothers' famous flights, ornithopter research was mostly abandoned in favor of the more promising fixed-wing designs. Certain inventors such as Alexander Lippisch and Percival Spencer [2], however, continued developing models for manned and unmanned flight. Cox et al. worked on piezoelectrically actuated ornithopters with novel control schemes [3]. Finally, DeLaurier successfully built a piloted ornithopter and it flew for 14 s in 2006 [4].

A major obstacle to these pursuits has been the counterintuitive aerodynamic behavior of air around moving wings. Wagner [5] and Theodorsen [6] provided an early mechanism for fixed wings in flutter which has been applied by Jones [7] and DeLaurier [8] to accelerating wings. DeLaurier expanded this to cover the root-flapping wings of an ornithopter. That particular characterization is not difficult to implement for practical computation, and it was compared to actual flapping models with success [8,9]. Whether these methods, based on potential flow, are applicable for micro-air-sized vehicles is undetermined, however.

Entomologists and their cohorts working along a different vein have produced aerodynamic models for insect flight. Weis-Fogh and Jensen [10] applied quasi-steady analysis to analyze hovering insects and produced estimates of the lift and drag coefficients of insect wings. Ellington [11] produced a seminal review of insect-flight aerodynamics, mainly using this quasi-steady viewpoint, but with disturbingly high lift coefficients [12]. Computers became sufficiently powerful during the 1990s to allow two-dimensional time-dependent simulation of oscillating airfoils, led by Wang [13] and Russel and Wang [14]. Several 3-D approaches have also been attempted, albeit with a deficiency in accuracy requiring excessive computing time [13]. During this same time, experiments with mechanical wings have been developed to confirm these computational results for a hawk moth and a fruit fly [15], providing helpful visualization as well. Finally, Wang et al. [16] provided a quasi-steady approximation to their numerical simulations that will be further outlined in this paper.

The problem this paper addresses is that of designing a flapping-wing micro air vehicle: in particular, we want to know how it will fly. There exists little literature that provides a practical outline of the ornithopter design process, which includes predicting such things as vehicle configuration, power requirements, stress–strain analysis for the wings and internal structures, proper wing kinematics for maneuvering, actuator selection, and control system design. Here, we concentrate on dynamically modeling ornithopter flight as a function of selected vehicle geometry, control surface deflections, and wing kinematics. The authors have developed a coupled vehicle dynamics/aerodynamics model for longitudinal flight, which is then used to analyze flight dynamics patterns for predetermined wing kinematics functions, and is used to study trim states for sustained forward flight. Similar to the work of Taylor et al. [17], this work studies the stability of flapping-wing vehicles. Taylor applies continuous-time linear time-invariant averaging methods to study dragonfly flight dynamics, and this produces eigenvalue estimates for the averaged system. Because of the much faster time scale of dragonflies' flapping frequency compared with the characteristic time scales of the dragonflies' flight dynamics modes, the flight dynamics are largely uncoupled from the flapping dynamics and the averaging method works excellently. This paper studies larger ornithopters where the flapping frequency's time scale is closer to the characteristic time scale of the vehicle's dynamic modes, thereby making averaging methods less applicable. To analyze an ornithopter with significantly more motion coupling between body dynamics and wing beating, the authors use a method of limit-cycle analysis (accounting for nonlinear dynamics effects) to produce eigenvalue calculations independent of flapping time scales.

II. Modeling

To design ornithopter features to provide enough lift to overcome weight and to produce enough thrust to overcome drag, the authors have modeled a standard ornithopter configuration consisting of a fuselage, a pair of symmetric wings, and a standard airplane empennage (Fig. 1). The empennage contains control authority over the rudder and elevator, but there are no ailerons on the wings (the longitudinal assumption makes this moot, but this will be more significant with further analyses, as DeLaurier's ornithopter relies on yaw-roll coupling in place of ailerons). The ornithopter simulated by the authors has a wingspan of 0.72 m. The vehicle's mass is 0.12 kg, with each wing 0.0079 kg. The vehicle's center of mass and tail's quarter-chord line are 9.5 and 30.5 cm rear of the wings' quarter-chord line. The wings' planform is 803 cm^2 and the tail's planform is 209 cm^2.

The wings are attached to the fuselage via a hinged joint, allowing two rotational degrees of freedom (Fig. 1). The first, and most prominent, is the vertical plunging (or heaving) of the wings $\phi(t)$. This is the characteristic oscillating dihedral motion of avian forward flight, requiring the most energy of all the controlled motions. All other kinematics on the ornithopter are referred to this oscillatory

Received 13 August 2007; revision received 4 January 2008; accepted for publication 8 January 2008.

*Graduate Student, Sibley School of Mechanical & Aerospace Engineering, 226 Upson Hall. Student Member AIAA.
†Associate Professor, Sibley School of Mechanical & Aerospace Engineering, 224 Upson Hall. Member AIAA.

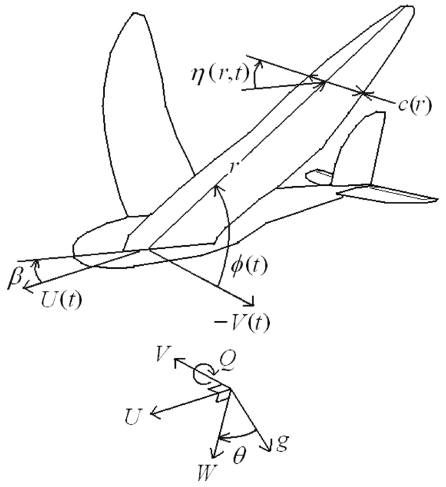

Fig. 1 Ornithopter prototype schematic. The ornithopter for our study has two flapping wings and standard vertical and horizontal stabilizers.

motion and its phase. The second degree of freedom is a wing twist about the quarter-chord line $\eta(r, t)$. It is noted in the literature that this oscillating motion is approximately 90 deg out of phase from plunging [8]. These motions are given as follows:

$$\phi(t) = C_h \cos(\omega t) + \phi_0 \tag{1}$$

$$\eta(r, t) = C_t \frac{r}{R} \cos(\omega t + \phi_{\eta_0}) \tag{2}$$

where ω is in radians per second. Realistic wing structures for flapping flight are flexible and respond aeroelastically to the fluid forces and to their own inertial loads. In birds, this is quite dramatic as the feathers perform gross shape change due to their flexibility and the supporting structure's flexibility. Our ornithopter wing model assumes that the wings are rigid except for predefined twisting and plunging angles.

Fluid dynamics around the vehicle are a different issue. As mentioned earlier, full computational fluid dynamic simulations of the time-dependent, three-dimensional, Navier–Stokes equations is not practical for vehicle dynamics simulation. Impracticalities include unknown fluid initial conditions and boundary conditions, unknown required resolution to realize the salient features of the flow, and of course the enormous scaling difficulties with three-dimensional flowfield discretization. Thus, simplifications are in order. Because two-dimensional computational fluid dynamics (CFD) simulations have shown veracity to robotic flapping experiments [15], the authors have adopted calculations relying on wing sections rather than the entire wing. This is similar to the methods of DeLaurier [8] for larger models. This must be further simplified, however, if computations are to be rapid enough to enable the analysis of vehicle dynamics. A set of algebraic formulas for forces on an airfoil, designed to match the general behavior of the 2-D CFD results, are thus employed as derived by Wang et al. [16]. This method sacrifices precision for computability, but this is warranted by our application: generation and analysis of ornithopter flight trajectories.

Force formulas are derived from three main modes: circulation, added mass, and viscous drag. Circulation for flapping wings is approximated as such (from Andersen et al. [18]):

$$\Gamma(r) = -2C_L c(r) \frac{uv}{|\mathbf{v}|} + 2C_R c(r)^2 \dot{\eta} \tag{3}$$

The first of the two terms is the translation-induced circulation, the general mechanism for lift generation in fixed-wing aircraft. The second term, absent from standard fixed-wing aircraft aerodynamics, depends on rotation rate, and is derived from the lift on a zero angle-of-attack pitching plate [18].

Added mass is a term that compensates for the acceleration of the mass of the air near the airfoil, an effective additional inertial load affecting the total force on the wing. Had this been a time-dependent simulation of the Navier–Stokes equations, this term would not exist as a separate term but would be embedded in the momentum of the general fluid motion. Quasi-steady approximations, however, require corrections to adhere to the true unsteady flow patterns.

Finally, the viscous drag force is the most intuitive force among these outlined here. It estimates the effect of the viscous boundary layer, but also separated pressure-driven drag:

$$d\mathbf{F}_v = \rho r c(r) |\mathbf{v}| \{C_D(0)\cos^2\alpha + C_D(\pi/2)\sin^2\alpha\} dr \begin{bmatrix} u \\ v \end{bmatrix} \tag{4}$$

Wang [12] gives the forces on the airfoil as

$$dF_x = \left\{ \left(\frac{m_w c(r)}{\bar{c}} + m_{22} \right) \frac{r}{R} v\dot{\eta} - \rho r v \Gamma - m_{11} \frac{r}{R} \dot{u} \right\} dr - dF_{v_x} \tag{5}$$

$$dF_y = \left\{ -\left(\frac{m_w c(r)}{\bar{c}} + m_{11} \right) \frac{r}{R} u\dot{\eta} + \rho r u \Gamma - m_{22} \frac{r}{R} \dot{v} \right\} dr - dF_{v_y} \tag{6}$$

$$d\tau = \left\{ (m_{11} - m_{22}) \frac{r}{R} vu - \frac{I_a}{R} \ddot{\eta} + \frac{c(r)}{4} \rho v \Gamma \right\} dr - d\tau_v \tag{7}$$

These equations can then be compared with those of DeLaurier [8] in simulation, thus providing confidence in their correlation to larger-scale aircraft; at the intermediary between insect flight and a human-piloted ornithopter, the models largely agree. The simulation is run with a given set of sinusoidal wing kinematics and a given steady air velocity, and the resulting total lift is shown in Fig. 2. In this instance, the force equations act on the ornithopter whose fuselage is constrained rigidly in space as though it were resting on a sting in a wind tunnel.

The differences in the simulated results arise from the differences in modeling assumptions between the two methods. Wang's model [18] used herein, approximates the vortex shedding effects through empirical matching of a series of force coefficients. These originate in the study of unsteady flow features such as leading-edge vortex and spanwise flow which are found in insect wings. DeLaurier's model [8] uses classical theoretical aerodynamics developed within the regime of piloted aircraft. It accounts for the unsteady vortex effects of flapping through Theodorsen functions, a result of applied potential theory to unsteady wing motions [7]. Opposed to Wang's insect motivated corrections [18], this uses the long-studied aerodynamics of conventional aircraft and aeroelastic wings. In the actual calculations of forces from these models, however, many of the terms appear the same; there are components proportional to wing velocity, rotation rate, translational acceleration, and rotational acceleration in both.

Ornithopter trajectories are governed by the Newton–Euler equations of rigid body motion in the local frame:

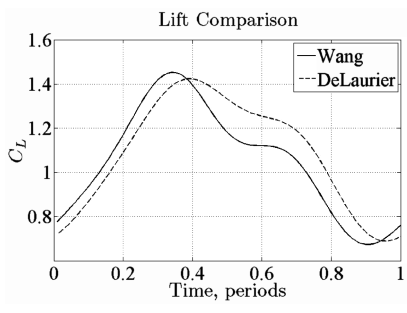

Fig. 2 Wang's method [12] vs Delaurier's method [8].

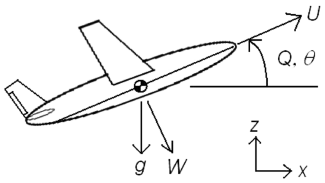

Fig. 3 Schematic of dynamic variables in the longitudinal plane.

$$\mathbf{F} = m\frac{\mathrm{d}\mathbf{v}_c}{\mathrm{d}t}\bigg|_B + m(\boldsymbol{\omega} \times \mathbf{v}_c) \tag{8}$$

$$\mathbf{M} = \frac{\mathrm{d}\mathbf{H}}{\mathrm{d}t}\bigg|_B + \boldsymbol{\omega} \times \mathbf{H} \tag{9}$$

The equations are simplified for longitudinal motion here:

$$\dot{U} = -QW - g\sin\Theta + F_x/m \tag{10}$$

$$\dot{W} = QU + g\cos\Theta + F_z/m \tag{11}$$

$$\dot{Q} = M_y/I_y \tag{12}$$

$$\dot{\Theta} = Q \tag{13}$$

$$\dot{x} = U\cos\Theta + W\sin\Theta \tag{14}$$

$$\dot{z} = U\sin\Theta - W\cos\Theta \tag{15}$$

where U, W, Q, and Θ are illustrated in Fig. 3, and F_x, F_z, and M_y include all aerodynamic forces and moments in the body frame of reference. The term m represents the mass of the fuselage, and ρ is the air density. These coordinates can be normalized using the following parameters: a characteristic length $l_0 = 2m/\rho S$, a characteristic velocity $V_0 = \sqrt{gl_0}$, a characteristic time $t_0 = \sqrt{l_0/g}$, and a characteristic rotation rate $\omega_0 = 2\pi/t_0$. These parameters are chosen due to their independence from kinematic variables like flapping frequency or amplitude.

These expressions, combined with the preceding aerodynamic formulas, are easily implemented in software for simulation under given input kinematics (Fig. 1). The authors use MATLAB [19] exclusively, as in previous work [20].

III. Trim Analysis

We want to find the trim state of the ornithopter as a function of its elevator angle, wing twist amplitude, plunging amplitude, and flapping frequency. Unfortunately, for ornithopters, there is no state where U, W, Q, and Θ are constant in time, the usual definition of trim for a fixed-wing aircraft. Forces on an ornithopter can be periodic, however, and so we define ornithopter trim as a limit cycle with the same frequency as the flapping input.

A limit cycle [21] $\boldsymbol{x}^*(t)$ of a dynamic system (16) is an isolated periodic solution of the system's equations of motion:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}), \qquad \mathbf{x} \in R^n \tag{16}$$

$$\mathbf{x}(t+T) = \mathbf{x}(t), \quad \forall\ t \tag{17}$$

Limit cycles are invariant; any trajectory starting on a limit cycle remains there for all time. Also, limit cycles are generally stable or unstable; trajectories starting near a limit cycle either approach the limit-cycle trajectory or diverge from it. The typical method to quantify the stability of a limit cycle is to analyze the Poincaré map, a mapping of the intersection of the trajectory with a hyperplane transverse to the limit cycle:

$$\mathbf{x}(k+1) = P[\mathbf{x}(k)] \tag{18}$$

Selection of such a plane is not unique; the plane can intersect the limit cycle anywhere along the trajectory. Thus, $\mathbf{x}(k)$ and $\mathbf{x}(t)$ represent the same state, but $\mathbf{x}(k)$ only occurs at discrete instants in time. In this mapping, a point near the limit cycle will map to another point on the hyperplane, and the point where the limit cycle intersects the plane is an equilibrium point $\mathbf{x}^*(k)$ in discrete time:

$$\mathbf{x}^*(k+1) = P[\mathbf{x}(k)] = \mathbf{x}^*(k) \tag{19}$$

and thus can be analyzed for stability. The Jacobian of the Poincaré map taken at the equilibrium is called the monodromy matrix, and its eigenvalues, called the Floquet multipliers, determine the linear stability of the limit cycle. Indeed, the monodromy matrix is the discrete-time state matrix in the linearization of the Poincaré map:

$$\{\mathbf{x}(k+1) - \mathbf{x}^*(k+1)\} = M\{\mathbf{x}(k) - \mathbf{x}^*(k)\} \tag{20}$$

The Floquet multipliers have interesting properties:

1) They are independent of the hyperplane choice for the Poincaré map, because they are a property of the limit cycle.

2) At least one Floquet multiplier, called the trivial multiplier, is identically one. Notably, the eigenvector associated with the trivial multiplier points along the limit-cycle trajectory. A limit cycle is stable if all its Floquet multipliers, save the trivial multiplier, have complex modulus less than one.

The periodically-forced ornithopter system can be seen as an autonomous system if a set of differential equations can be found that integrate to the required forcing functions. The forcing functions are augmented to the state vector of the system and treated as dynamic states. Because, for our ornithopter, the flapping is governed by sinusoids, finding their differential equations is trivial; they are a linear combination of sines and cosines in the form $\ddot{x} + \omega^2 x = 0$. Interestingly, the resulting monodromy matrix for the forced ornithopter system is in block form, and so the trivial multipliers associated with the forcing can be ignored in stability calculations. The block containing the interesting system properties, which is the block illustrated next, thus will not contain the trivial multiplier.

The monodromy matrix can be calculated from an initial value problem. The nonlinear system can be linearized about the limit cycle and a state transition matrix can be calculated as follows. Then, the linearized system is

$$\frac{\mathrm{d}\{\mathbf{x}(t) - \mathbf{x}^*(t)\}}{\mathrm{d}t} = \left[\frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}}\bigg|_{\mathbf{x}=\mathbf{x}^*(t)}\right]\{\mathbf{x}(t) - \mathbf{x}^*(t)\} \tag{21}$$

The solution to this linear time-varying ordinary differential equation is

$$\{\mathbf{x}(t) - \mathbf{x}^*(t)\} = \boldsymbol{\Phi}(t, t_0)\{\mathbf{x}(t_0) - \mathbf{x}^*(t_0)\} \tag{22}$$

where $\boldsymbol{\Phi}(t, t_0)$ is the state transition matrix, the solution of the differential equation:

$$\frac{\mathrm{d}\boldsymbol{\Phi}(t, t_0)}{\mathrm{d}t} = \left[\frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}}\bigg|_{\mathbf{x}=\mathbf{x}^*(t)}\right]\boldsymbol{\Phi}(t, t_0), \qquad \boldsymbol{\Phi}(t_0, t_0) = I \tag{23}$$

If $(t - t_0)$ equals the period of flapping, then the state transition matrix equals the monodromy matrix. Thus, the monodromy matrix can be calculated through numerical integration of this initial value problem. The challenge is to calculate the limit-cycle trajectory and its Jacobian. State transition matrices have the semigroup property

$$\boldsymbol{\Phi}(t_2, t_0) = \boldsymbol{\Phi}(t_2, t_1)\boldsymbol{\Phi}(t_1, t_0) \tag{24}$$

which allows the monodromy matrix to be expressed as a product of state transition matrices of the system around subsections of the limit

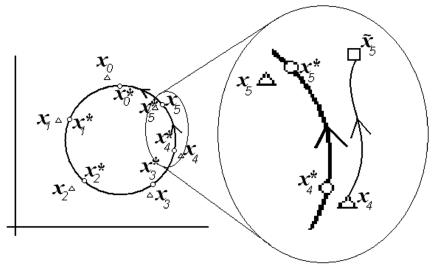

Fig. 4 Schematic of multiple-shooting algorithm.

cycle. This becomes valuable for the multiple-shooting method described here.

The limit cycle is broken into a series of segments (Fig. 4) in time, delineated by a set of time points and coordinates [22]:

$$0 < t_1 < t_2 < \cdots < t_{m-1} < T \tag{25}$$

$$\{\mathbf{x}(0), \mathbf{x}(t_1), \ldots, \mathbf{x}(t_{m-1}), \mathbf{x}(t_m)\} = \{\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_{m-1}, \mathbf{x}_m\} \tag{26}$$

and let the numerical solution of the dynamic system between time points be denoted as

$$\{\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, \ldots, \tilde{\mathbf{x}}_{m-1}, \tilde{\mathbf{x}}_m\} \tag{27}$$

then the periodicity of the limit cycle is enforced by the system of equations

$$\begin{cases} \tilde{\mathbf{x}}_1 - \mathbf{x}_1 = 0 \\ \tilde{\mathbf{x}}_2 - \mathbf{x}_2 = 0 \\ \vdots \\ \tilde{\mathbf{x}}_m - \mathbf{x}_m = 0 \\ \mathbf{x}_m - \mathbf{x}_0 = 0 \end{cases} \tag{28}$$

which are satisfied when $\mathbf{x}_i = \mathbf{x}_i^*$, and these equations can be solved using Newton's method. Knowing that the integrated trajectory termini $\tilde{\mathbf{x}}_i$ depend on the guess of the initial condition $\mathbf{x}_{i-1}$ only, and the guesses $\mathbf{x}_i$ depend only on themselves, the iterated equations become

$$\begin{bmatrix} \frac{\partial \tilde{\mathbf{x}}_1}{\partial \mathbf{x}_0}, & -I, & 0, & \cdots & 0, & 0 \\ 0, & \frac{\partial \tilde{\mathbf{x}}_2}{\partial \mathbf{x}_1}, & -I, & \cdots & 0, & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0, & 0, & 0, & \cdots & \frac{\partial \tilde{\mathbf{x}}_m}{\partial \tilde{\mathbf{x}}_{m-1}}, & -I \\ -I, & 0, & 0, & \cdots & 0, & I \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x}_0 \\ \Delta \mathbf{x}_1 \\ \Delta \mathbf{x}_2 \\ \vdots \\ \Delta \mathbf{x}_{m-1} \\ \Delta \mathbf{x}_m \end{bmatrix} = - \begin{bmatrix} \tilde{\mathbf{x}}_1 - \mathbf{x}_1 \\ \tilde{\mathbf{x}}_2 - \mathbf{x}_2 \\ \vdots \\ \tilde{\mathbf{x}}_m - \mathbf{x}_m \\ \mathbf{x}_m - \mathbf{x}_0 \end{bmatrix} \tag{29}$$

where the Jacobians $\partial \tilde{\mathbf{x}}_i / \partial \mathbf{x}_{i-1}$ are the state transition matrices $\mathbf{\Phi}(t_{i-1}, t_i) = M(i-1)$, calculated about the guessed trajectories. When the guessed trajectories are correct, they connect to form the limit cycle and the Jacobians can combine to form the state transition matrix for the entire circuit:

$$\mathbf{\Phi}(t_m, t_{m-1})\mathbf{\Phi}(t_{m-1}, t_{m-2}) \cdots \mathbf{\Phi}(t_2, t_1)\mathbf{\Phi}(t_1, t_0) = \mathbf{\Phi}(t_m, t_0) = M \tag{30}$$

where M is the aforementioned monodromy matrix for the circuit. Also note that monodromy matrices for each point calculated in the multiple-shooting algorithm can as easily be calculated by changing the order of multiplication. Thus, we have produced a method for linearizing and analyzing the system's internal dynamics, but the next step is to study the effect of parameter changes on those dynamics.

General nonlinear systems of the form of Eq. (31) can be linearized around a nominal control input u at any time step [as in Eq. (32)]:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \tag{31}$$

$$\mathbf{x}(k+1) = M(k)\mathbf{x}(k) + G(k)\mathbf{u}(k) \tag{32}$$

where $G(k) = \tilde{G}(t_{k+1})$, the input matrix, is the solution to this initial value problem:

$$\frac{\mathrm{d}\tilde{G}(t)}{\mathrm{d}t} = \left[\frac{\partial \mathbf{f}(t, \mathbf{x}, \mathbf{u})}{\partial \mathbf{x}}\bigg|_{\mathbf{x}(t),\mathbf{u}(t)}\right]\tilde{G}(t) + \left[\frac{\partial \mathbf{f}(t, \mathbf{x}, \mathbf{u})}{\partial \mathbf{u}}\bigg|_{\mathbf{x}(t),\mathbf{u}(t)}\right], \quad \tilde{G}(t_k) = 0 \tag{33}$$

which can be solved numerically, and $\tilde{G}(t)$ is a dummy integration variable. [Or if in discrete time, $\mathbf{x}(k+1) = \mathbf{f}\{\mathbf{x}(k), \mathbf{u}(k)\}$, then $G(k) = \partial \mathbf{f}\{\mathbf{x}(k), \mathbf{u}(k)\}/\partial \mathbf{u}(k)$.] What this means is that the general nonlinear controlled system (31) can be analyzed using the tools of control theory for linear discrete-time systems. Of particular interest is to apply that theory to design a feedback controller to stabilize the system, that is, to create a rule for parameter changes that makes the system stable. The additional property of periodicity in the trajectory is advantageous; the input matrices become periodic

$$G(k+m) = G(k) \tag{34}$$

and all controller algorithms can be computed a priori.

Another challenge is calculation of the Jacobians

$$\frac{\partial \mathbf{f}(t, \mathbf{x}, \mathbf{u})}{\partial \mathbf{x}}\bigg|_{\mathbf{x}(t),\mathbf{u}(t)}$$

and

$$\frac{\partial \mathbf{f}(t, \mathbf{x}, \mathbf{u})}{\partial \mathbf{u}}\bigg|_{\mathbf{x}(t),\mathbf{u}(t)}$$

The underlying equations are difficult to write out, and the symbolic rendition of their derivatives is practically impossible to set down. One solution to this is to use finite difference differentiation, but this operation's error is unknown. An alternative is automatic differentiation, a method that computes derivatives of a function while the function's value is computed. One particular implementation for MATLAB, ADMAT [23], creates a new class of variable with two components: the functional value and its derivative. As MATLAB runs through the order of operations upon the first component, a running derivative for each operation is computed and combined into the second component using a table of derivatives for each basic MATLAB function and the chain rule of differentiation. This produces derivative values to machine precision. For example, if given this function

$$f(x) = x \sin(\omega x + \phi) \tag{35}$$

the order of operations would be as shown in Table 1. Thus, automatic differentiation calculates the derivative simultaneously with the function value.

IV. Dynamic Analysis on an Ornithopter

Using the preceding methodologies, we will now analyze the flight dynamics of an ornithopter. Through trial and error, several limit cycles were found in these analyses, with one eventually selected for further analysis. The same limit cycle will be used in the following

Table 1 Automatic differentiation example

Operation no.	Function	Derivative
1	$x_1 = \omega^* x + \varphi$	$x_1' = \omega$
2	$x_2 = \sin(x_1)$	$x_2' = x_1'^* \cos x_1$
3	$x_3 = x^* x_2$	$x_3' = x_2 + x^* x_2'$
4	$f(x) = x_3$	$\mathrm{d}f(x)/\mathrm{d}x = x_3'$

control design section to produce a stable trim for ornithopter flight. The first item of interest is the convergence of the multiple-shooting algorithm outlined in Eqs. (25–29). Figure 5 shows that, for this example, the Newton method converges quadratically in as few as four iterations, to produce a precise list of points on the limit cycle. Other examples show similar convergence rates, depending on the initial condition.

The limit cycle illustrated here (Fig. 6) is the example used here for stability analysis and control design. It represents a near straight and level trajectory (Fig. 7). The monodromy matrix for this limit cycle, as calculated using Eq. (30), is:

$$M = \begin{bmatrix} 1.0565 & -0.16208 & -0.29621 & -2.0939 \\ -0.37531 & 0.33912 & 0.63510 & 0.34042 \\ -0.34225 & 0.55420 & 1.1198 & 0.15016 \\ -0.039648 & 0.079971 & 0.20947 & 1.0105 \end{bmatrix} \tag{36}$$

where its discrete-time eigenvalues are

$$\lambda = 1.9782, 0.014836, 0.76642 \pm 0.36764i \tag{37}$$

hence, the system is unstable (Fig. 8).

Figure 6 presents the dynamic states of the ornithopter during three periods of one limit cycle. These are the trim values for each state, from which the unstable system would diverge. Figures 7 and 8 are results from one simulation: Fig. 7a shows the longitudinal trajectory of the ornithopter in free flight, and Fig. 7b shows the values of the states during the flight. Figure 8 shows the deviation of the ornithopter's dynamic states from their trim values on the limit cycle as the ornithopter diverges. The ornithopter pitches up near the 2 s mark, enters a pitching tumble, and the vehicle begins to dive. This is all caused by the instability in the system.

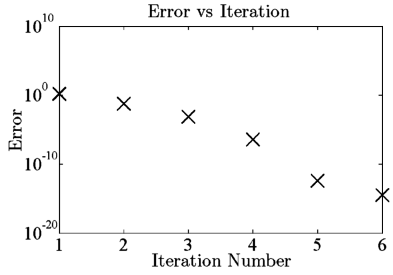

Fig. 5 Multiple-shooting convergence. Error is defined as the 2-norm of the right side of Eq. (27).

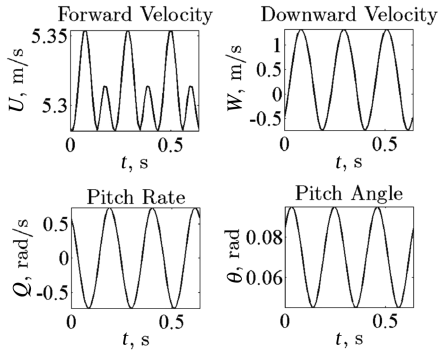

Fig. 6 Limit-cycle illustration.

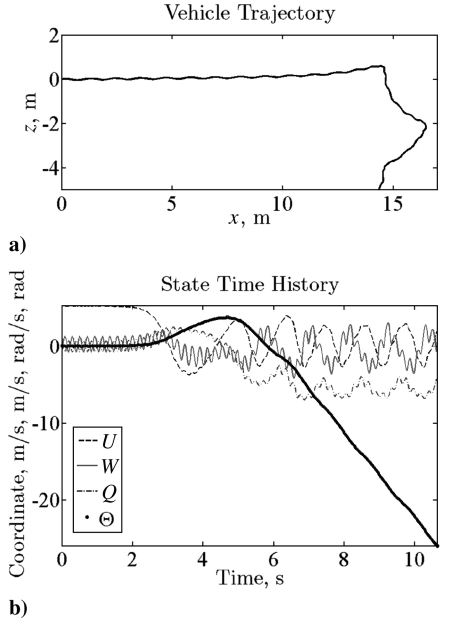

Fig. 7 Flight trajectory of ornithopter: a) inertial coordinates, b) dynamic states.

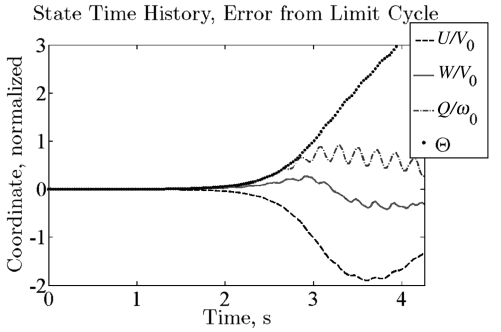

Fig. 8 Error dynamics for a perturbed ornithopter, normalized coordinates.

Modal analysis provides further illustration of the system. The unstable eigenvalue $\lambda_1 = 1.9782$ has the eigenvector $v_1 = [-0.68, 0.42, 0.57, 0.19]^T$, meaning that this mode excites all the states with a similar magnitude. It has a doubling time of 0.21618 s (1.016 times the flapping period). This is consistent with the unstable mode in Taylor, whose dragonfly simulations had one unstable eigenvalue exciting all the states significantly with a doubling time on the order of the flapping period [17]. The stable real eigenvalue $\lambda_2 = 0.014844$ has the eigenvector $v_2 = [0.060, 0.90, -0.43, 0.022]^T$, which has a disproportionate effect on W and Q, similar to the short period mode without oscillation. It has a time constant of 0.0505 s. Finally, the stable complex eigenvalues $\lambda_{3,4} = 0.76644 \pm 0.36764i$ has eigenvectors $v_{3,4} = [0.83, 0.067 \pm 0.17i, 0.43 \pm 0.25i, 0.052 \mp 0.19i]^T$, having a disproportionate effect on U and Q, which is similar to a traditional phugoid mode with a damping ratio of 0.34 and a natural frequency of 2.16 rad/s. The settling time for this mode ($t_s = 4/\zeta\omega_n$) is 5.42 s.

V. Conclusions

We have constructed equations of motion for an ornithopter with periodic wing motions. The aerodynamics are a function of the vehicle states, and the physics reveals complex nonlinear relationships for those states. Proposed was a root solving method which used limit cycles that characterized the periodic motion in an ornithopter's states without a priori estimates of the vehicle's trim velocities, or even proof of their existence.

Limit cycles are merely a feature of the phase space in ornithopter flight dynamics, that when found herald powerful implications. They represent the only analog to a dynamic equilibrium in this time-explicit (nonautonomous) dynamic system; with flapping wings, there is no constant force and only a trivial case would maintain constant states in the system. We found a numerical approximation of a limit cycle to computer precision allowing for a study of the system's stability with the same precision as calculating the eigenvalues of the Jacobian of an equilibrium point. In our analysis, the estimation of the system's eigenvalues is independent of the flapping frequency and amplitude, which figure largely in averaging methods. The least precise part of the dynamic analysis is now the aerodynamic model. The Floquet algorithm provides a starting point to find other trim points (different flapping frequencies, wing twist amplitudes, or elevator deflections), necessary for a more comprehensive vehicle design. Future possibilities are to use this methodology as a basis to solve for optimal kinematics for low-power trajectories, high-speed trajectories, and unsteady maneuvers such as takeoff, landing, and lateral movements.

Acknowledgments

The authors would like to acknowledge the National Science Foundation Integrative Graduate Education and Research Training fellowship program, and would like to acknowledge advice from John Guckenheimer.

References

[1] Moon, F. C., "*The Machines of Leonardo Da Vinci and Franz Reuleaux: Kinematics of Machines from the Renaissance to the 20th Century*," Springer, Dordrecht, The Netherlands, 2007, pp. 255–257.

[2] DeLaurier, J. D., and Harris, J. M., "Study of Mechanical Flapping-Wing Flight," *Aeronautical Journal*, Vol. 97, Oct. 1993, pp. 277–286.

[3] Cox, A., Monopoli, D., Cveticanin, D., Goldfarb, M., and Garcia, E., "Development of Elastodynamic Components for Piezoelectrically Actuated Flapping Micro-Air Vehicles," *Journal of Intelligent Material Systems and Structures*, Vol. 13, No. 9, 2002, pp. 611–615.

[4] Black, D., "It Flies! Aviation History is Made by the 'Flapper'," *Toronto Star*, Toronto, 9 July 2006, p. A03.

[5] Wagner, H., "Über die Entstehung des dynamischen Auftriebes von Tragflügein," *Zeitschrift für Angewandte Mathematik und Mechanik*, Vol. 5, No. 1, Feb. 1925, pp. 17–35. doi:10.1002/zamm.19250050103

[6] Theodorsen, T., "General Theory of Aerodynamic Instability and the Mechanism of Flutter," NACA TR 496, 1935.

[7] Jones, R. T., "Unsteady Lift of a Wing of Finite Aspect Ratio," NACA TR 681, 1940.

[8] DeLaurier, J. D., "Aerodynamic Model for Flapping-Wing Flight," *Aeronautical Journal*, Vol. 97, No. 7, April 1993, pp. 125–130.

[9] DeLaurier, J. D., "Development and Testing of an Efficient Ornithopter Wing," *Aeronautical Journal*, Vol. 97, No. 8, May 1993, pp. 153–162.

[10] Weis-Fogh, T., and Jensen, M., "Biology and Physics of Locust Flight, 1: Basic Principles in Insect Flight. A Critical Review," *Proceedings of the Royal Society of London, Series B: Biological Sciences*, Vol. 239, No. 667, 1956, pp. 415–458. doi:10.1098/rstb.1956.0007

[11] Ellington, C. P., "Aerodynamics of Hovering Insect Flight," *Philosophical Transactions of the Royal Society of London, Series B: Biological Sciences*, Vol. 305, No. 1122, 1984, pp. 1–181. doi:10.1098/rstb.1984.0049

[12] Wang, Z. J., "Dissecting Insect Flight," *Annual Review of Fluid Mechanics*, Vol. 37, Jan. 2005, pp. 183–210. doi:10.1146/annurev.fluid.36.050802.121940

[13] Wang, Z. J., "Two Dimensional Mechanism for Insect Hovering," *Physical Review Letters*, Vol. 85, No. 10, 2000, pp. 2216–2219. doi:10.1103/PhysRevLett.85.2216

[14] Russell, D., and Wang, Z. J., "Cartesian Grid Method for Modeling Multiple Moving Irregular Objects in 2D Incompressible Viscous Flow," *Journal of Computational Physics*, Vol. 191, No. 1, 2003, pp. 177–205. doi:10.1016/S0021-9991(03)00310-3

[15] Dickinson, M. H., Lehmann, F. O., and Sane, S. P., "Wing Rotation and the Aerodynamic Basis of Insect Flight," *Science*, Vol. 284, No. 5422, 1999, pp. 1954–1960. doi:10.1126/science.284.5422.1954

[16] Wang, Z. J., Birch, J., and Dickinson, M., "Unsteady Forces and Vorticity Field in Hovering Flight: Two Dimensional Computations vs Robotic Wing Experiments," *Journal of Experimental Biology*, Vol. 207, No. 3, 2004, pp. 449–460. doi:10.1242/jeb.00739

[17] Taylor, G. K., Bomphrey, R. J., and Hoen, J., "Insect Flight Dynamics and Control," *44th AIAA Aerospace Sciences Meeting and Exhibit*, AIAA 2006-32, Jan. 2006.

[18] Andersen, A., Pesavento, U., and Wang, Z. J., "Analysis of Transitions Between Fluttering, Tumbling and Steady Descent of Falling Cards," *Journal of Fluid Mechanics*, Vol. 541, Oct. 2005, pp. 91–104. doi:10.1017/S0022112005005847

[19] MATLAB, Software Package, Ver. 7.2.0.232, MathWorks, Natick, MA, 1984–2007.

[20] Dietl, J. M., and Garcia, E., "Kinematic Variation and Modeling for Design in Ornithoptic Flight," *American Society of Mechanical Engineers International Mechanical Engineering Congress and Exposition*, American Society of Mechanical Engineers IMECE2005-82035, Nov. 2005.

[21] Strogatz, S. H., *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering*, Westview Press–Perseus Books, Cambridge, MA, 1994, Chaps. 7–8.

[22] Lust, K., "Improved Numerical Floquet Multipliers," *International Journal of Bifurcation and Chaos in Applied Sciences and Engineering*, Vol. 11, No. 9, 2001, pp. 2389–2410. doi:10.1142/S0218127401003486

[23] Coleman, T. F., and Verma, A., "ADMIT-1: Automatic Differentiation and MATLAB Interface Toolbox," *ACM Transactions on Mathematical Software*, Vol. 26, No. 1, 2000, pp. 150–175. doi:10.1145/347837.347879