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

Swarming/Flocking and Collision Avoidance for Mass Airdrop of Autonomous Guided Parafoils

Anthony J. Calise*
Atair Aerospace, Inc., Collegeville, Pennsylvania 19426
and
Daniel Preston†
Atair Aerospace, Inc., Brooklyn, New York 11231

DOI: 10.2514/1.28586

Swarming and flocking are two words used interchangeably to refer to modeled flight that is biologically inspired by the flight of flocking birds and swarming insects. The capability of biological flight systems to autonomously maneuver, track, and pursue evasive targets in a cluttered environment is vastly superior to current engineered systems. Our effort seeks to mimic the behavior of biological systems to improve the design of engineered guidance and control systems for autonomous unmanned aerial vehicles and, in particular, parafoil-based systems. This paper details our development from concept through simulation through the first successful implementation on fleets of five unmanned aerial vehicle paragliders.

Nomenclature

a	=	weighting parameter in the flocking guidance law
C_D	=	drag coefficient
C_L	=	lift coefficient
$c_{j,i}$	=	ϕ_i center corresponding to input $\bar{x}_j$
D	=	drag
dh	=	difference in altitude
$d(t)$	=	disturbance
$e(t)$, $e_f(t)$	=	error signal and filtered $e(t)$
F	=	neural network adaptation gain
g	=	gravity
h	=	altitude
h_g	=	altitude where V_g is specified
h_t	=	elevated target altitude
K_e	=	e-modification gain in the adaptive law
K_f, K_ψ	=	guidance flocking and heading gains
K_p, K_i	=	stability augmentation system proportional and integral gains
K_{rm}	=	reference model gain
L	=	lift
m	=	mass
Q	=	dynamic pressure
R	=	range
$\mathbf{R}_r$	=	relative position vector
r_0, r_1	=	flocking guidance range parameters
S	=	reference area
T	=	sampling time interval
t, t_{go}	=	time and time to go
V	=	velocity
$\bar{V}$	=	average velocity
V_g	=	glide velocity at $h = h_g$
$\mathbf{V}_r$	=	relative velocity vector
W	=	weight
W_i	=	neural network ith basis function weight
W_{int}	=	wind integral along ψ_d
w_x, w_y	=	horizontal wind components
$\hat{w}_{p\psi d}$	=	estimated wind perpendicular to ψ_d
$\hat{w}_{\psi d}$	=	estimated wind along ψ_d
x	=	north position relative to the target site
$\bar{x}$	=	neural network input vector
y	=	east position relative to the target site
y_m	=	reference model output
γ, γ_g	=	flight-path angle and γ at $h = h_g$
δ, δ_c	=	servo position and servo command
$\delta\psi$	=	heading error
$\delta\psi_f$, $\delta\psi_t$	=	flocking and target site $\delta\psi$
δ_{ad}	=	adaptive component of the servo command
ζ	=	damping ratio
μ	=	bank angle
μ_{ss}	=	bank angle in steady state
ρ, ρ_g	=	air density and ρ at $h = h_g$
σ	=	basis function width
τ_{rm}	=	reference model time constant
ϕ_i	=	neural network basis function element
ψ	=	heading
$\dot{\psi}_c$	=	heading rate command
ψ_d	=	desired heading
ω_n	=	natural frequency

Introduction

GUIDED parafoils provide a means of achieving precision delivery of supplies from air to ground. Figure 1 shows an image of the system that is the subject of this paper. The Onyx is a two-parachute system that is static line deployed from an aircraft at altitudes of up to 35,000 feet. An elliptical profile is used for the initial deployment and autonomous guidance to the target site. An onboard flight computer determines its position and heading based on its integrated global positioning system (GPS) and inertial navigation sensors. The system then executes a calculated flight pattern until it reaches its preprogrammed altitude and position, where a second nonguided parachute deploys just before landing for a soft touchdown.

The system can operate in an environment in which multiple units are deployed in the same airspace. This is accomplished using a collision avoidance algorithm and formation flight algorithms, which herein we refer to as flocking. The main purposes of flocking are to

Presented as Paper 6477 at the Guidance, Navigation and Control Conference, San Francisco, CA, 15–18 August 2005; received 27 October 2006; revision received 22 October 2007; accepted for publication 22 October 2007. Copyright © 2008 by Anthony J. Calise. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission. Copies of this paper may be made for personal or internal use, on condition that the copier pay the $10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 0731-5090/08 $10.00 in correspondence with the CCC.

*Senior Analyst, 8009 Woodgate Circle, and Professor, School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332-0150; acalise@ae.gatech.edu. Fellow AIAA.

†Chief Executive Officer, 499 Van Brundt Street, Suite 3B. Member AIAA.

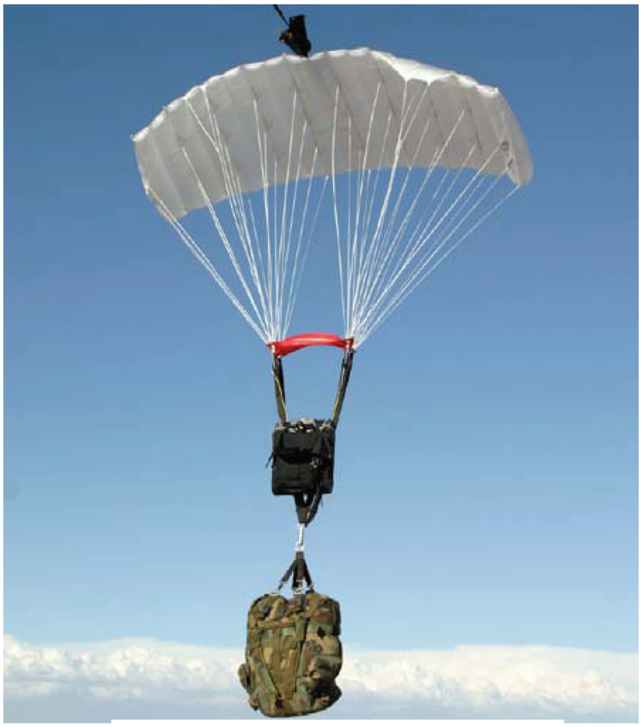

Fig. 1 Onyx 500 system.

minimize the possibility of collisions and to have the individual units arrive at the target site in an organized manner. This is done without controlling their respective relative positions, which could place an excessive burden on the ultimate goal of reaching the target site. Therefore, we view the need for flocking as a preventive measure, and collision avoidance as a maneuver of last resort. A second requirement is that the algorithms operate in a fully decentralized fashion, so that there is no need for a supervisory controller.

Standard approaches for formation control include leader–follower, virtual structure, and behavior-based structure approaches. In the leader–follower approach [1,2], one vehicle is designated as a leader and the remaining vehicles as followers. The leader sets a nominal trajectory for the formation to follow and may cooperate with the followers in regulating range. In the virtual structure approach, the entire formation is treated as a single entity [3,4]. The desired motion is assigned to this single entity, the virtual structure, which traces out trajectories for each member in the formation to track. In behavior-based approaches [5,6], several desired behaviors are prescribed for each vehicle and the final control is derived from a weighting of the relative importance of each behavior. Because coordination is with respect to a central agent in the leader–follower and virtual structure approaches, the formation control lacks robustness. The behavior-based approaches are decentralized and are significantly easier to implement. However, they are difficult to analyze mathematically, and formation convergence to the desired configurations is not guaranteed.

Although imperfectly understood, the flocking behavior of birds, schooling behavior of fish, and even studies of swarming insects have provided inspiration for concepts of coordinated multivehicle operation [7]. Reynolds introduced a model that suggests flocking is the combined result of three simple steering rules that each agent follows independently [8]. In this model, each agent can access the whole scene's geometric description, but flocking requires that it react only to flock mates within a certain small neighborhood. Reynolds' rules were validated in a graph-theoretic and Lyapunov stability analysis framework [9,10]. The convergence properties on individual agent velocity vectors and relative distances were shown. Olfati and Murray [10] also provided a framework for addressing the splitting, rejoining, and squeezing maneuvers for flocks in the presence of multiple obstacles.

Another aspect in guidance and flight control design is the requirement for adaptation in both the guidance functions and in the stability augmentation system (SAS) design. This is due to the need to design a system that enables users to adopt and deploy equipment in the most practical manner. That is, there is a need to self-adapt to a variety of drop conditions, payload shapes, and wing loadings that are not known beforehand. There is also a need to be able to drop multiple systems in the same airspace without confliction. Additionally, high-glide systems must allow for the possibility of multiple targets to be reached in a single sortie. Adaptive algorithms permit varying cargo weights and shapes to be rigged under a given system without requiring this data as a part of the mission planning. An adaptive SAS design also permits systems to fly correctly with asymmetrically rigged payloads or to correct for damage-induced asymmetries midflight. Our approach to adaptive SAS design employs a method for augmenting a linear controller design [11]. The linear controller in this case is a baseline SAS design that is much like the SAS design used in other systems [11–13]. This approach is ideally suited for dealing with the nonlinear response characteristics of parafoils. It also employs an approach for protecting the adaptive process from the effects of servo position and rate limits [14].

In this paper, we describe our approach to the modeling and design of adaptive guidance and control algorithms, including collision avoidance and flocking. There are a variety of high-fidelity models described in the literature [15–18]. Our approach has been to use a reduced-order model sufficient for the purposes of guidance and flight control design. There are three advantages derived from this approach. First, there are only a few parameters that need to be determined from flight tests to tune the model for a given application. Second, this eliminates the need for employing complex algorithms for system identification and parameter estimation [19–21]. And third, flocking strategies with multiple parafoils can more quickly be evaluated, and a considerable amount of insight is gained in the design phase. Simulated and flight-test results are presented that illustrate some of these advantages.

The paper begins with a description of our approach to modeling and describes the manner in which guidance and the baseline SAS was designed. This is followed by a description of the flight tests that have been performed thus far to demonstrate our design, the last of which consisted of a formation flight of five unmanned aerial vehicles that occurred in December 2004. Although at this point we have not yet completed our implementation and testing of a fully adaptive SAS, we describe its design and the present simulated results indicative of its anticipated performance.

Parafoil Modeling and Guidance and Control Law Development

Equations of Motion

The force components in the wind frame are depicted in Fig. 2. The equations of motion for gliding flight over a flat and nonrotating Earth and in a constant wind field can be expressed as

$$\begin{aligned} \dot{V} &= (-D - W\sin\gamma)/\mathrm{m} & \dot{\gamma} &= (L\cos\mu - W\cos\gamma)/\mathrm{mV} \\ \dot{\psi} &= L\sin\mu/mV\cos\gamma & \dot{x} &= V\cos\gamma\cos\psi + w_x \\ \dot{y} &= V\cos\gamma\sin\psi + w_y & \dot{h} &= V\sin\gamma \end{aligned} \tag{1}$$

The equations for equilibrium flight are obtained by setting $\dot{V} = \dot{\gamma} = 0$, which reduces the first three equations in Eq. (1) to

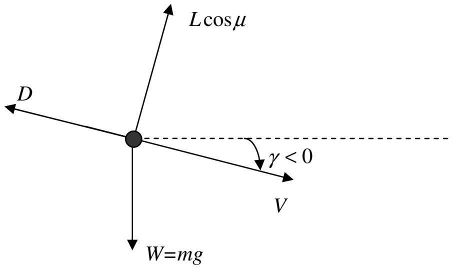

Fig. 2 Force components in the wind frame.

$$D = -W \sin\gamma \qquad L\cos\mu = W\cos\gamma \qquad \dot{\psi} = g\tan\mu/V \tag{2}$$

Combining the first two equations in Eq. (2), it follows that

$$\tan\gamma = \frac{D}{L\cos\mu} = -\frac{C_D}{C_L\cos\mu} \tag{3}$$

where $C_L = L/QS$, $C_D = D/QS$, and $Q = \rho(h)V^2/2$. In equilibrium glide ($\dot{V} = \dot{\gamma} = \dot{\psi} = 0$), it follows from Eqs. (2) and (3) that

$$\tan\gamma_g = -C_D/C_L \tag{4}$$

Comparing Eqs. (3) and (4), the flight-path angle in turning flight can be expressed in terms of γ_g as

$$\tan\gamma = \tan\gamma_g/\cos\mu \tag{5}$$

We can also derive a relationship between the equilibrium glide velocity (V_g) and the equilibrium velocity during turning flight, which we have already referred to as V. Both are altitude dependent, but for modeling purposes it is useful to think in terms of specifying V_g for a single altitude (h_g) and then computing V for any arbitrary altitude and bank angle condition. From the second equation in Eq. (2), we have $\rho(h_g)V_g^2SC_L = 2W\cos\gamma_g$ in gliding flight and $\rho V^2SC_L = 2W\cos\gamma/\cos\mu$ in turning flight. Therefore,

$$V = V(h,\mu) = \sqrt{\left\{\frac{\rho_g\cos\gamma(\mu)}{\rho(h)\cos\gamma_g\cos\mu}\right\}}V_g \tag{6}$$

where ρ_g denotes $\rho(h_g)$. Note that γ in Eq. (6) is regarded as a function of μ due to the relationship in Eq. (5). If we also specify the glide ratio (GR), then this uniquely determines γ_g needed in Eq. (6) from the relationship

$$\tan\gamma_g = -(1/\text{GR}) \tag{7}$$

Thus, the kinematics of the reduced-order model obtained by setting $\dot{V} = \dot{\gamma} = 0$ are completely characterized in terms of V_g, h_g, and GR. If of interest, the lift and drag coefficients can be obtained from the aforementioned relations by using Eq. (7) to establish γ_g, then using the second equality in Eq. (2) and the relation in Eq. (4):

$$C_L = 2W\cos\gamma_g/\rho_gV_g^2 \qquad C_D = -C_L\tan\gamma_g \tag{8}$$

Although this is not required for simulation purposes, it is useful for parafoil design.

The parafoil simulation consists of integrating the following set of equations:

$$\begin{aligned} &\dot{x} = V\cos\gamma\cos\psi + w_x \qquad \dot{y} = V\cos\gamma\sin\psi + w_y \\ &\dot{h} = V\sin\gamma + \omega_h \qquad \dot{\psi} = g\tan\mu/V \\ &\ddot{\mu} = -2\zeta w_n\dot{\mu} - \omega_n^2(\mu - \mu_{\text{ss}}(\delta)) \end{aligned} \tag{9}$$

where ζ and ω_n are the damping ratio and natural frequency associated with the bank angle transient response, and μ_{ss} is the steady-state bank angle that is a function of the servo position (δ). The expressions for γ and V in terms of m are given in Eqs. (5) and (6). The use of Eq. (9) for simulation purposes amounts to assuming that the parafoil is always in a state of symmetric equilibrium flight, $\dot{V} = \dot{\gamma} = 0$, and zero sideslip.

The last term in Eq. (9) is used to describe the transitory bank angle response to a servo displacement. This is characterized in terms of ζ, ω_n, and the servo characteristic ($\mu_{\text{ss}}(\delta)$), which in general is a nonlinear function that relates the *steady-state* bank angle to the servo displacement. A typical parafoil characteristic is shown in Fig. 3. It is made up of a dead zone, a low-gain linear region, a very high-gain region (not necessarily linear as depicted here), and what is represented here as a saturation region in which the parafoil actually enters an uncontrollable spiraling motion. The points of transition between these regions are mainly dependent on wing loading and variations in canopy construction. It is difficult to ascertain where the transitions lie beforehand unless the transitions have been calibrated for a given canopy by flight testing at different wing loadings. Even then, these transitions can vary as the canopy is reused over time. In particular, note that the bank angle is not zero when the servo stroke is zero. Even if it were zero, canopies can have a bias of their own that produces a nonzero turn rate when at a zero bank angle. From a modeling perspective, the net effect is the same and is well represented by the model in Fig. 3. The dead zone is largely due to slack in the control lines that may exist at the zero reference position for the servo stroke. This dead zone can be quite large depending on the wind loading.

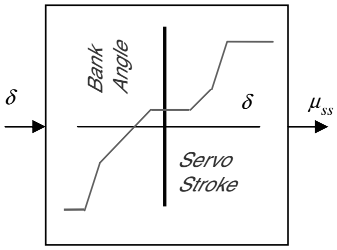

Fig. 3 Typical servo characteristic.

We refer to the slope in the low-gain region as the *parafoil gain*. The combination of the natural frequency, damping ratio, and parafoil gain values ultimately determine the gains that are used in the guidance and SAS design. These values are what distinguish one class of parafoils from another, together with the values of V_g, h_g, and GR. On the other hand, within a class of parafoils there can be significant variation in the servo characteristic due variations in the gross weight and canopy construction. These variations are taken into account in the adaptive portion of our design.

As described, there are only a few parameters that need to be identified from flight tests to make use of the simulation model in Eq. (9). Estimating V_g, h_g, and GR from GPS data is straightforward; however, it does require doing a complete 360-deg turn to correct the estimates of V_g and GR for wind. Figure 4 illustrates how we estimate the natural frequency, damping ratio, and servo characteristics from flight-test responses. In this flight test, step changes in the servo position were commanded and the heading rate of the inertial velocity vector was estimated from the horizontal components of the GPS-indicated velocity. We then tuned the values of ω_n, ζ, and the parameters of the servo characteristic in simulation to match as best we could the measured response. The simulated responses are the smooth lines in this figure, whereas the somewhat noisy response in the heading rate corresponds to the estimates obtained from the GPS data. Note that the values of ω_n and ζ are best observed when there is a

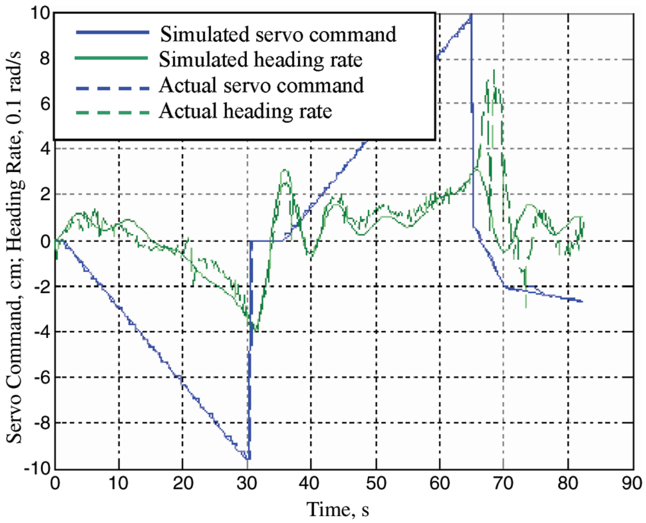

Fig. 4 Comparison of the Onyx 2200 simulated and flight-test data.

large change in the servo command from the maximum negative value back to zero at $t = 30$ s and from the large positive value back to zero at $t = 60$ s. The parafoil gain is best observed from the incremental changes in the servo command. The left and right dead zone and canopy bias are adjusted to capture the regions in which there is little or no response to small step changes in the servo and to approximate the nonzero turn rate at the zero servo stroke.

In summary, the design transition from one weight class to another involves first obtaining the step response data of the type illustrated in Fig. 4, then estimating the parameters ω_n and ζ and the servo characteristics by tuning the simulation that results from the numerical integration of the differential equations in Eq. (1), and finally computing the proportional and integral (PI) gains used in the guidance and SAS designs as a function of these parameters. The simulation can then be used to predict the guidance and control system performance under varying wind conditions and canopy characteristics and to calculate the envelope of release conditions from which a mission can be completed, which in turn can be used for mission planning purposes. The use of the reduced-order model in Eq. (9) greatly facilitates this process.

Guidance Law Equations

The guidance law consists of the following flight phases:

1) Deploy, turn toward the target site, and fly straight for a specified time period to estimate V_g and γ_g.

2) Turn for a specified time period with $\delta = \delta_1$ to estimate the rate of descent and the turn rate.

3) Turn toward the target site.

4) a) Use the estimates from the first two phases to compute a desired altitude for the current range to the target.

b) Execute a turn until the parafoil arrives near, but not below, the desired altitude.

c) Command weaves as needed to maintain the altitude error below a prescribed limit.

d) Command a turn rate proportional to the heading error.

5) Spiral when over the target site to dissipate any excess altitude that might remain.

During the first flight phase, V_g and γ_g are estimated using a simple average of the GPS sensor data. The value for V at the target altitude is also estimated based on Eq. (6), which for level flight simplifies to $V = \sqrt{\rho_g/\rho(h)\}}V_g$. During the second flight phase, the rate of descent is estimated using a simple averaging process. This calculation only relies on altitude data, which can be provided by either the GPS or an altimeter. During the third flight phase, a turn is executed until the parafoil arrives at the desired heading (ψ_d) to within a prescribed limit. The desired flight heading is given by

$$\psi_d = \tan^{-1}(-y, -x) \tag{10}$$

This is done to align the parafoil with the target heading before initiating the fourth flight phase.

The fourth flight phase is the most complex and consists of a number of subphases. During subphase 4a, an average velocity ($\bar{V}$) along the *ideal* glide slope is first computed by averaging the values $V(h)$ and $V(h_t)$, where h_t denotes the elevated target altitude (a specified altitude above the target site). This is then used to compute a desired altitude (h_d) so that flying at the value of γ_g, at a speed equal to $\bar{V}$ and at $\psi = \psi_d$, the parafoil will arrive at the target site at an altitude of $h = h_t$. The calculation is based on the following relationships:

$$-\dot{R} = \bar{V}\cos\gamma_g\cos\delta\psi + \hat{w}_{\psi d}(h) \qquad \dot{h} = \bar{V}\sin\gamma_g \tag{11}$$

where R is the current horizontal range to the target point (obtained from the GPS data), $\hat{w}_{\psi d}$ is the estimated wind component along ψ_d, and $\delta\psi$ is the difference between ψ_d and the ideal heading of the velocity relative to the air mass when flying along the *ideal* glide slope at $\psi = \psi_d$. This is given by the relation

$$\sin\delta\psi = \hat{w}_{p\psi d}/\bar{V}\cos(\gamma_g) \tag{12}$$

where $\hat{w}_{p\psi d}$ is the wind component perpendicular to ψ_d. Taking the ratio of the two expressions in Eq. (11)

$$-\frac{\mathrm{d}R}{\mathrm{d}h} = \frac{\bar{V}\cos\gamma_g\cos\delta\psi + \hat{w}_{\psi d}(h)}{\bar{V}\sin\gamma_g} \tag{13}$$

and integrating

$$-\int_R^0 \mathrm{d}R = \frac{\cos\gamma_g\cos\delta\psi}{\sin\gamma_g}\int_{h_{\mathrm{des}}}^{h_t} dh + \int_{h_{\mathrm{des}}}^{ht} \hat{w}_{\psi d}(h)\,\mathrm{d}h/\bar{V}\sin\gamma_g \tag{14}$$

Carrying out the indicated integrations

$$h_d = h_t - \{R\tan\gamma_g + [w_{\mathrm{int}}(h_t) - w_{\mathrm{int}}(h_d)]/\bar{V}\cos\gamma_g\}/\cos\delta\psi \tag{15}$$

where $w_{\mathrm{int}}(h)$ denotes a wind integral along ψ_d

$$w_{\mathrm{int}}(h) = \int_0^h \hat{w}_{\psi d}(h)\,\mathrm{d}h \tag{16}$$

Because ψ_d is variable from one guidance update to the next, the wind integrals are precomputed along the x and y directions and tabulated as a function of h as a part of the wind data set. Then the wind integral along ψ_d in Eq. (16) is computed from the tabulated x and y integrals by adding their respective components along ψ_d.

Equation (15) represents a nonlinear algebraic relation for h_d, which is solved using a fixed-point iteration until the change in h_d is less than 3 m. The maximum number of iterations is also limited, as a precaution. The iteration process has never failed to converge in over 10^6 trials and usually converges within three iterations. As a final step, the value of h_d is biased upward so that the bias linearly decreases as a function R, but never below a minimum specified value. This range-dependent bias is introduced to compensate for errors in our knowledge of winds. This precaution is taken to ensure that the parafoil does not fall below the unbiased desired glide slope, because it is impossible to recover from such a condition. Once subphase 4a is initiated, it is recomputed at every guidance update for the remainder of the flight.

During subphase 4b, a number of 360-deg turns are executed such that the parafoil arrives nearest, but always above, the value of h_d computed in subphase 4a. The number of turns is computed based on the estimate of altitude rate and turn rate obtained in the second phase. The phase is initiated together with subphase 4a and continues until the computed number of turns are executed. The value of h_d from subphase 4a is also biased to further ensure against ending up below h_d after completing the turns, which can be caused by the servo rate limits and the fact that h_d may be increasing due to the effect that winds have during the 360-deg turns. Other than possibly decreasing the number of 360-deg turns, this bias does not affect the subsequent subphases, whereas the bias introduced in subphase 4a does.

Subphase 4c commences after the completion of subphase 4b and continues for the remainder of the flight. This subphase is intended to initiate and execute a weaving maneuver whenever the altitude error ($h - h_d$) exceeds a prescribed limit. The altitude error increases with passing time, and the weave maneuver is intended to reset it to a small value. The growth in the altitude error is primarily due to the gradual reduction in the bias used in the calculation of h_d (see subphase 4a). Recall that this bias was introduced to prevent the parafoil from falling below the *ideal* glide slope. In this way, even if the present altitude should temporarily fall below h_d, it will eventually rise above h_d due to the gradual removal of the bias term in the calculation of h_d. Thus, the weave maneuver is needed to prevent too great a growth in the altitude error. This rate of growth depends on the prevailing wind conditions and can be significantly reduced by a large headwind. The calculation of the weave maneuver entails first commanding a new heading that is perpendicular to the desired heading and then maintaining this heading until the altitude error is reduced to 50% of its original value. At this point, the heading command is reset to the calculated ideal value needed to intercept the target site. This procedure does not require GPS coordinates (meaning that the inertial measurement unit, or IMU, heading can be used as an alternate to the GPS if the GPS signal is lost during the maneuver).

Subphase 4d is used to regulate the heading so that ψ is maintained close to ψ_d. It is executed throughout the fourth flight phase, except during subphases 4b and 4c. The commanded turn rate is limited in magnitude and used as the input to the SAS. The desired heading is biased off of the target site to allow for the terminal maneuver in the fifth phase. This fifth phase consists simply of removing the bias in ψ_d at a point where the turn rate command without the bias exceeds a threshold value. The primary objective of the fifth phase is to reduce the terminal error by performing a brief terminal spiral maneuver.

Baseline SAS Design

The baseline SAS is designed as a PI controller that regulates the heading rate that is commanded by the guidance system. The PI gains depend primarily on the parafoil gain and the bandwidth set for the design. The achievable bandwidth depends on a variety of factors that include the inherent damping in the parafoil response (ζ), the actuator rate limits, and the suspension system vibrations that corrupt the measurement of heading rate. The design is then evaluated using a simulation of the vehicle dynamics, as defined by Eq. (9), combined with the model for $\mu_{ss}(\delta)$, depicted in Fig. 3. A first-order model for each servo's position response (with a position and rate limit) is included to model the servo dynamics. To implement the SAS, it is necessary to feed back the heading rate, which is available from the IMU. The indicated heading rate is first passed though two identical first-order filters to remove the noise and harmonic terms due to oscillations in the suspension lines. The suspension model consists of a 0.3-rad/s amplitude, 2-Hz sinusoid added to the heading rate. The overall dynamics of the SAS loop is depicted in Fig. 5. The main factor that limits the bandwidth of the SAS design is the phase shift introduced by the discrete filters. An alternative architecture in which bank angle feedback is used in place of heading rate that promises to remedy this problem is suggested in Calise and Preston [22]. Performance is also sensitive to nonlinearity in the parafoil response and the servo rate limit. Thus, the ability to handle a wide range of payloads without tuning the controller design to the payload requires that this basic design be made adaptive to the payload and environmental conditions. The manner in which this design is augmented with an adaptive element is covered in the last section.

Simulated Results for a Single System

The parafoil parameter values are $W = 476$ N, $S = 3.77$ m^2, GR $= 3.0$, $V_g = 11.5$ m/s at $h_g = 3$ km, $\zeta = 0.8$, and $\omega_n = 2.5$ rad/s. The guidance and SAS parameters are $K_\psi = 0.5$ s^{-1}, $K_p = 10$ cm/rad, and $K_i = 1.0$ cm/rad, where K_ψ is the proportional guidance gain (heading rate is commanded proportional to heading error). The guidance commands are limited to 0.5 rad/s. The servos are modeled as first-order systems with a time constant of 0.2 s and a rate limit of 40 cm/s. The parafoil is assumed to produce 0.16 rad of bank per centimeter of servo stroke in the low-gain region and to have 0.1 rad of bank bias when the left and right servos are at zero stroke. A dead zone of 2.6 cm of servo travel was also modeled in each servo.

Figure 6 illustrates a typical simulated behavior for a single parafoil system. Note that for this release condition a long descending spiral occurs in the fourth flight phase to arrive near the glide slope. The weaving maneuvers as the parafoil approaches the target site are also evident.

Flocking Behaviors and Collision Avoidance

Flocking Behaviors

The intent of flocking is to bring a group of systems closer together without colliding and without specifying a rigid formation. To implement flocking, it is required that each air unit broadcasts its GPS coordinates to all of the other air units. Flocking behavior is not permitted after initiating a spiral down to the ideal glide slope. We first describe two behavior-based approaches. Consider the case of only two systems. Flocking will be defined as having each system steer a course relative to the other system with the objective of gradually reducing their relative range. The steering command is blended with a command to steer toward the target site as well. The exception is that, if any system is heading away from the target site, it is commanded only to steer toward the target site. The steer-toward-the-target-site command is based on the heading error ($\delta\psi_t$) generated by differencing the target site heading and the current system heading. This part of the steering command is identical to what was previously described as subphase 4d. The flocking command is generated by computing a collision heading for each system and using that heading to define a flocking heading error ($\delta\psi_f$). If a collision heading does not exist (the system is flying too slow relative to the other system), then a heading perpendicular to the line of sight (LOS) is commanded. The heading rate commands are generated proportional to these two heading errors and then blended according to

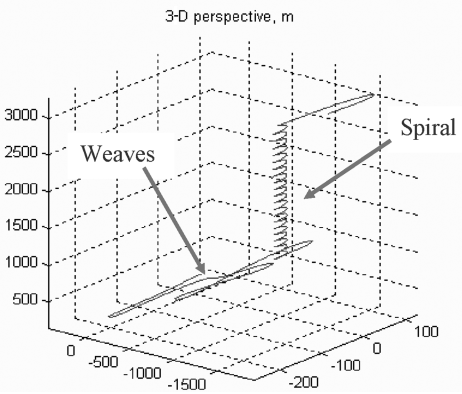

Fig. 6 Simulated behaviors and collision avoidance.

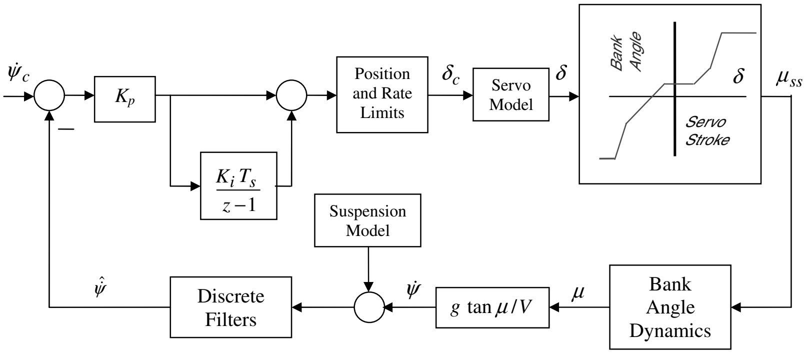

Fig. 5 Dynamics of the SAS loop.

$$\dot{\psi}_c = aK_f\delta\psi_f + (1-a)K_\psi\delta\psi_t \tag{17}$$

where the value of a depends on the range (R) to the other system:

$$a(R) = \left(\frac{R - r_o}{r_1 - r_o}\right)\bar{a} \tag{18}$$

The value of $\bar{a}$ is selected as a number between 0 and 1.0. The value of a is also limited between 0 and $\bar{a}$. This means that, when $R < r_o$, $a = 0$ (no flocking) and when $R > r_1 > r_o$, $a = \bar{a}$. The purpose of making the value of a range dependent is to reduce the likelihood of collisions.

For the case of multiple systems, the flocking strategy for each system is to steer relative to the system (other than itself) that is nearest the target site. An alternative behavior strategy is to steer toward the system farthest from the target site. This significantly increases the tendency to flock. Other systems are considered by the self-system to be members of the flock so long as they are guided and have not yet entered subphase 4b (spiraling down to their glide slope).

Figure 7 illustrates the simulated trajectories of five systems flying in a crosswind, in which each system pursues the system (other than itself) that is nearest the target site. The wind direction can easily be identified from the spiraling portion of these trajectories. In this case, there is a tendency for the parafoils to fly in a more or less V formation, with the leader being established by the fastest system. This case was simulated using $K_f = 0.1$ s^{-1}, $r_o = 46$ m, $r_1 = 107$ m, and $\bar{a} = 0.5$. Increasing $\bar{a}$ increases the tendency to form a tighter flock. Figure 8 illustrates the trajectories for each system pursuing the system (other than itself) that is farthest from the target site. Here there is a tendency to form a tighter flock, typically causing several collision avoidance maneuvers to occur near the end of the flocking phase, which is evident in the figure.

A third flocking strategy was investigated in which each vehicle follows a single leader, which is the vehicle closest to the target site that has not yet entered subphase 4b. This approach uses pure-pursuit guidance. In a pure pursuit, rather than command a collision heading (lead pursuit), a heading corresponding to the instantaneous LOS to the leader is commanded. This strategy does not require blending the commands for the followers ($a = 1$), because the leader flies directly toward the target site. Pure-pursuit guidance brings all of the parafoils into a line directly behind the leader. Its main disadvantage is that it relies on having a healthy leader. This form of guidance needs to be altered so that a new leader is automatically selected if it is determined that the current leader fails to guide properly. One way to do this is to monitor the heading error of the leader and require that the leader not only be closest to the target site, but also be either reducing or maintaining a small heading error. Figure 9 shows the results in the horizontal plane. None of the parafoils catch up with the parafoil in front of it because they all fly with the same velocity (they all have the same value of V_g). This may not be the case if the parafoils have different weights and/or different aerodynamic characteristics, and so this approach also suffers from this deficiency.

Two additional options have been implemented. Computationally, these are the simplest because they do not require any flight phases in which the glide slope or the rate of descent during a turn are estimated. They also provide the maximum range capability, because they maximize the energy when first arriving over the target site. Their main deficiency is that, when launched from short distances from the target site, the duration of the fifth phase can be very long, which is regarded as unacceptable for many missions. Also, collision avoidance over the target site becomes a problem. The first option is to fly straight to the target site without any spiral or weaving maneuvers and the second is to flock all the way to the target site. Figure 10 depicts the first option without collision avoidance implemented, and Fig. 11 depicts the second using follow-the-leader flocking behavior with collision avoidance implemented. Note the flowering of the trajectories over the target site due to the collision avoidance maneuvers that are executed. This can be avoided by introducing weaves to ensure a minimum separation range between adjacent units. However, this introduces added complexity and can

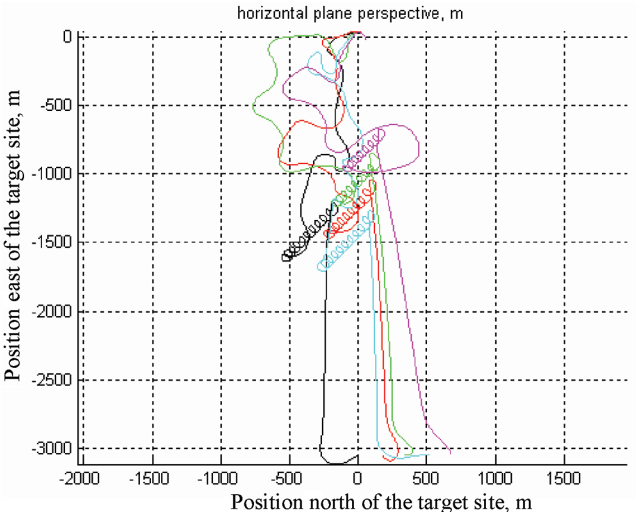

Fig. 7 Steer toward the system nearest the target site.

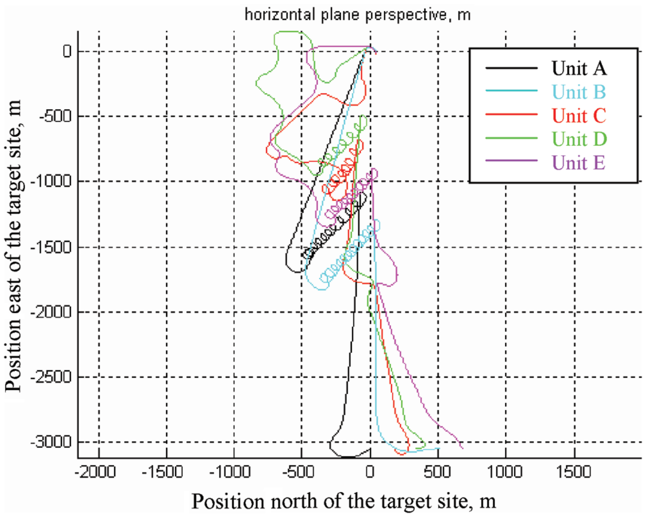

Fig. 8 Steer toward the system farthest from the target site.

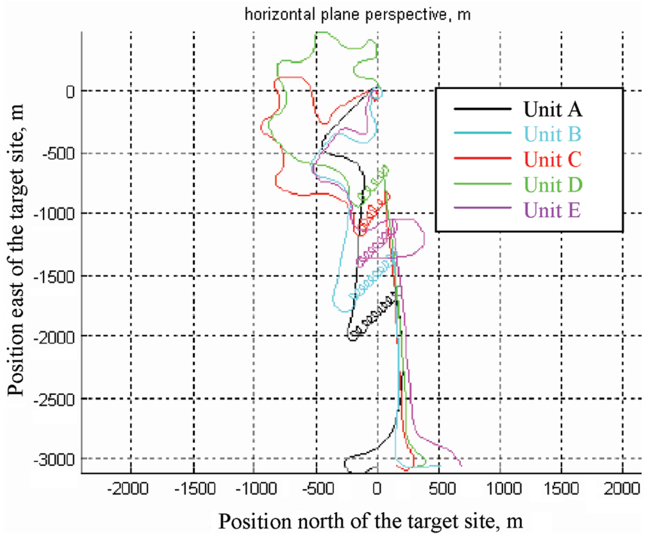

Fig. 9 Trajectories when using follow-the-leader flocking behavior.

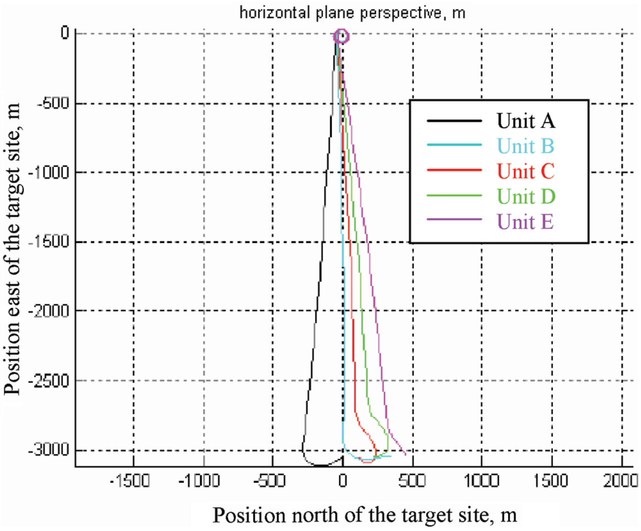

Fig. 10 Trajectories when flying straight to the target site.

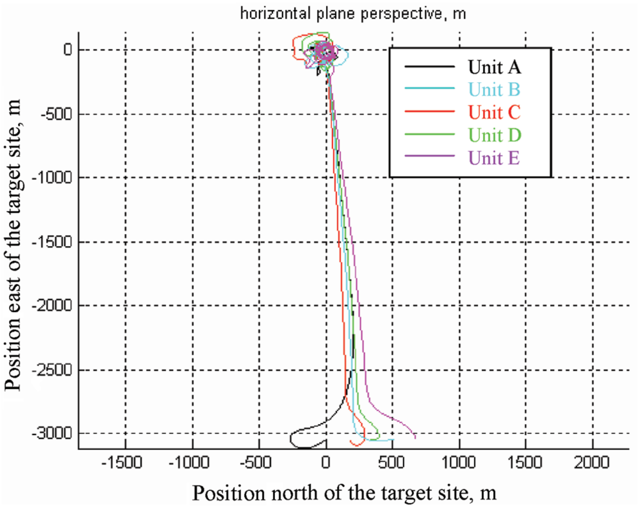

Fig. 11 Trajectories when following the leader all the way to the target site.

also introduce an undesirable response that propagates backward through the flock.

Several other interesting points to note are that the flight durations of all the flocking strategies are approximately the same, regardless of how the trajectories are flown. If the parafoils are launched close together with similar payloads, they are going to arrive in a bunch at the target site. All of the cases shown have collision avoidance maneuvering (see the next subsection) at the start involving parafoils 2, 3, and 4 due to their assumed initial headings.

Collision Avoidance

Collision avoidance is based on the concept of zero effort miss (ZEM) in the horizontal plane. At any time instant, ZEM is the miss distance that results at the point of closest approach if two systems were to continue to fly at their present speed and heading. If at any instant range rate is nonnegative, then ZEM is not defined. Otherwise, it can be shown that

$$\mathrm{ZEM} = |\mathbf{V}_r t_{\mathrm{go}} + \mathbf{R}_r| \tag{19}$$

and time to go is given by

$$t_{\mathrm{go}} = -\frac{\mathbf{R}_r^T \mathbf{V}_r}{|\mathbf{V}_r|^2} = \frac{-\dot{R}}{|\mathbf{V}_r|^2} \tag{20}$$

where $\dot{R}$ is the range rate. If $\mathrm{ZEM} < \mathrm{ZEM}_{\mathrm{min}}$, $t_{\mathrm{go}} < t_{\mathrm{go}_{\mathrm{min}}}$, and $dh < dh_{\mathrm{min}}$ where dh is the altitude difference, then a collision avoidance maneuver is executed. Onboard a given system, this test is performed for every transmission of position and velocity received from another system *that is presently under guided flight*. In addition, the lead angle to every system is also calculated. This is the angle between the velocity heading of the self-system and the LOS to the other system. If the lead angle is positive, or if the absolute value of the lead angle lies below a specified value, the system is commanded to execute a right turn at half of its maximum turn rate capability. This means that for near head-on collisions, both systems are commanded to turn right, whereas for other situations (e.g., glancing collisions) only the system to the right (looking down on the horizontal plane) is commanded to turn right. A collision avoidance maneuver, once initiated, is maintained until it is determined at a subsequent guidance update that at least one of the three aforementioned conditions is not active. Collision avoidance takes precedence over flocking behavior and all other behaviors, such as spirals and weaves. Collision avoidance is often required at launch and during spiraling maneuvers. At launch, when the parafoil is first deployed, an individual system quickly establishes an initial flight direction that is essentially random. This presents a serious problem in that the systems may not be staggered in either deployment time or deployment altitude. The limiting factor in collision avoidance is the ability to reliably predict that a collision is about to occur. The ZEM-based algorithm is particularly vulnerable when one or more vehicles are spiraling because it may not recognize that a collision is about to occur until it is too late. Several other approaches were developed that incorporate turn rate information. However, the more assumptions that are incorporated regarding the system behavior, the more vulnerable the approach becomes to the system doing something slightly different. Therefore, this effort was abandoned in favor of avoiding the situation that gives rise to the problem in the first place, which is described next.

The need for collision avoidance during the subphase 4b can be minimized by adding additional conditions that have to be satisfied before the initiation of this subphase is permitted. For example, the following additional criteria were required to be met before subphase 4b:

1) The parafoil must be the furthest from the terminal site among all of the parafoils that have not yet entered subphase 4b.

2) The horizontal range to any parafoil that has entered subphase 4b must be at least 150 m. This has the effect of increasing the duration of the third phase for some of the parafoils. This also orders the initiation of subphase 4b so that they are initiated starting from the back and progressing to the front of the flock.

3) All parafoils must spiral in the same direction.

Flight Testing

To prove out the guidance and SAS performance on "single systems," 134 drops were conducted. This culminated in a demonstration of 10 consecutive drops in June 2004. The average accuracy was 57 m, with the closest being 7 m and the farthest 84 m. The systems were dropped from a random offset of 1.6–34.8 km with a 3-km altitude and no knowledge of wind data. With this demonstrated, we went on to implement the developed flocking and collision avoidance algorithms. In December 2004, the flocking and collision avoidance algorithms were implemented onto a fleet of five systems, extensively bench tested, and then air dropped in two loads of five systems each. We chose to implement the flocking strategy of steering toward the system closest to the target site. With the new flocking and collision avoidance code implemented, the accuracy of the two drops averaged 170 m. However, due to extraordinarily severe and unfavorable wind conditions (greater than 15 m/s), all systems arrived at the terminal site at altitudes that did not permit them to enter the fifth phase and, as a consequence, their horizontal error components were larger than what would have been the case if they were permitted to use a terminal phase maneuver. Figure 12 shows an image taken in which four of the five systems are captured as they are approaching the landing site. The round, white landing

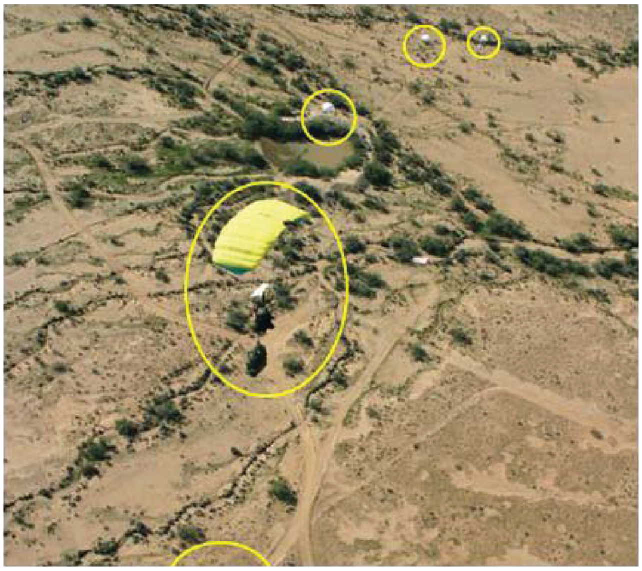

Fig. 12 Four of the five systems in frame during the landing stage.

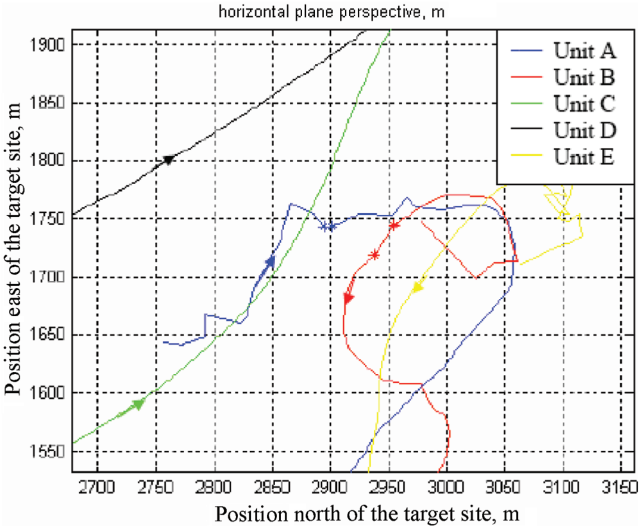

Fig. 13 Expanded view of the collision avoidance maneuver performed by unit A.

parachutes that are deployed for landing are visible in the upper portion of the image.

A collision avoidance maneuver did occur in this flight test. Evidence of the maneuver is shown in Fig. 13. This maneuver occurred shortly after the units were deployed and had a duration of 2 s. Unit A avoided a potential collision with unit B. The start and end of the maneuver are marked by the blue asterisks. The red asterisks indicate the corresponding positions of unit B, which at this point in time was in the air but had not initiated its guidance. None of the other units were in guided flight, and unit A had been in guided flight for only 2 s before the collision avoidance maneuver.

Adaptive SAS Design

The architecture of the adaptive SAS is illustrated in Fig. 14. The filter block that appears in the feedback path of the nominal SAS controller was used to reduce the presence of oscillations in the heading rate signal due to the suspension system and was a part of the nominal SAS design shown in Fig. 5. The "reference model" and "neural network" blocks were added to augment the nominal SAS design [11]. The reference model contains a first-order model for the parafoil dynamics, which is filtered in the same manner as the heading rate response of the actual parafoil. It also implements a method called "hedging," used to prevent the windup of the adaptive process in the presence of servo limits [14]. The reference model is use to define the error signal, $e(t)$, which is used to adapt the weights in the neural network (NN). The other input to the NN consists of a vector $(\bar{x})$ of known variables, consisting of the heading rate, the reference model output (y_m), and a one-step delayed value of the servo command (δ_c). The adaptive signal (δ_{ad}) is added to the existing SAS controller output as an augmenting signal, and its main purpose is to maintain $e(t)$ close to zero. Note that, with this architecture, external disturbances $(d(t))$ that affect the parafoil dynamics do not directly affect the reference model dynamics, so that in the process of reducing the error signal it also provides a level of disturbance rejection.

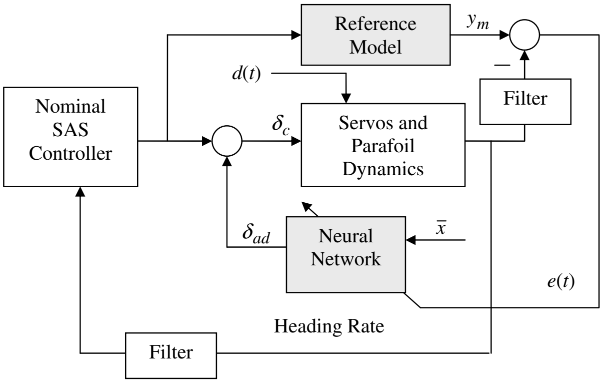

Fig. 14 Adaptive SAS architecture.

Neural Network

The NN output at each sample time (n) is computed as a weighted sum of radial basis functions, plus a term proportional to the error signal:

$$\delta_{\mathrm{ad}}(n) = -\sum_{i=1,\ldots,28} W_i\phi_i(\bar{x}(n)) + K_e e(n) \tag{21}$$

where

$$\phi_i = \exp\{-[(\bar{x}_1 - c_{1,i})^2 + (\bar{x}_2 - c_{2,i})^2 + (\bar{x}_3 - c_{3,i})^2/\sigma^2\} \tag{22}$$

for $i = 1, \ldots, 27$, and $\phi_{28} = 1$ is the network bias term. The elements of the input vector are normalized so that they approximately range between ±1. The parameters $c_{k,i}$ and σ are referred to as the center and width of the basis function. The centers are chosen so that they grid the input space.

Adaptation Law

The NN weights are adapted according to the following adaptation law [11]:

$$W_i(n+1) = W_i(n) - FTe_f(n)\phi_i(\bar{x}(n)) \tag{23}$$

where $e_f(n)$ is a filtered version of the error signal [11]. If any one of the weights should exceed a specified bound at any sample time, then all of the weights are adjusted using a projection operator so that largest weight is on its bound and the remaining weights are $\leq$ to their respective bounds [23].

Simulated Results

The network parameters are selected as follows: $F = 1.5$, $T = 0.05$ s, $\sigma = 0.5$, $K_e = 3\ \mathrm{s}^{-1}$, $\tau_{\mathrm{rm}} = 0.5$ s, and $K_{\mathrm{rm}} = 0.8$, where τ_{rm} and K_{rm} are the time constant and gain of the first-order reference model used in this application. Each of the network weights was bounded by 0.9. A first-order filter having a time constant of 1.0 s was used to filter the error signal. The first-order model for the servo response has a time constant of 0.05 s and a rate limit of 25 cm/s.

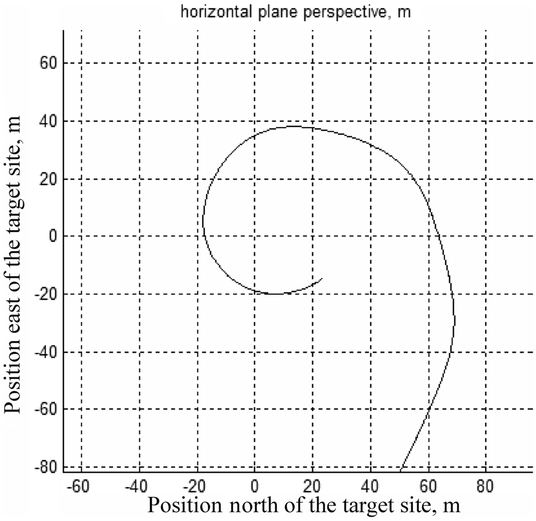

Fig. 15 Expanded view over the terminal site.

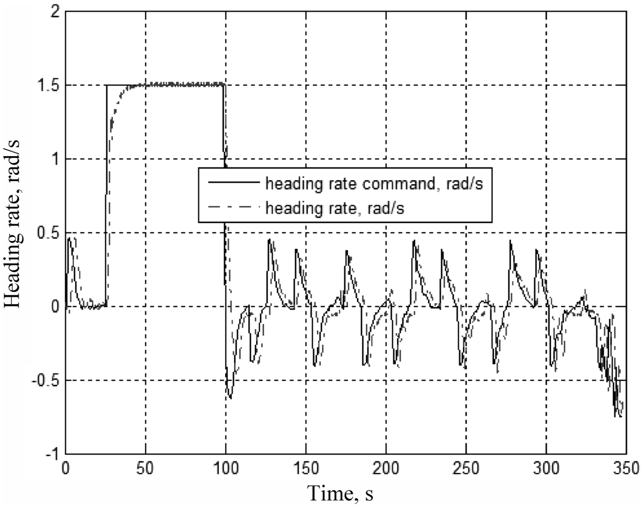

Fig. 16 Tracking performance with adaptation.

When the guidance commands are limited to a sufficiently low value, the parafoil responds in a nearly linear fashion. However, if this limit is increased beyond a critical value, the parafoil exhibits a highly nonlinear response in which the turn rate can no longer be regulated by the SAS. This is due to a sudden change in the control effectiveness that occurs, as depicted in Fig. 3. The slope was assumed to change from 20 to 80 deg/s of bank response when the servo position exceeds 7.7 cm. The response then saturates at 10.2 cm of servo position. Adaptation permits operation within the nonlinear response region and also permits a higher bandwidth in the nominal SAS design.

Figure 15 shows an expanded view of the trajectory near the terminal site. The terminal accuracy is significantly improved over that achievable with a turn rate limit of 0.5 rad/s. The same is true in the vertical plane (not shown). Figure 16 shows the excellent tracking performance achieved by the adaptive SAS during periods of both low and high turn rate commands. Figure 17 confirms the fact that the tracking performance of the nominal SAS design is poor by comparison. It exhibits a large oscillation during the spiral maneuver due to the nonlinear servo characteristic. In simulation we limit the bank angle response, but in flight tests a tumbling behavior occurs when high turn rates are commanded, which results in oss of control. Tracking performance is also improved when flying toward the target site due to the higher bandwidth afforded in the nominal SAS design.

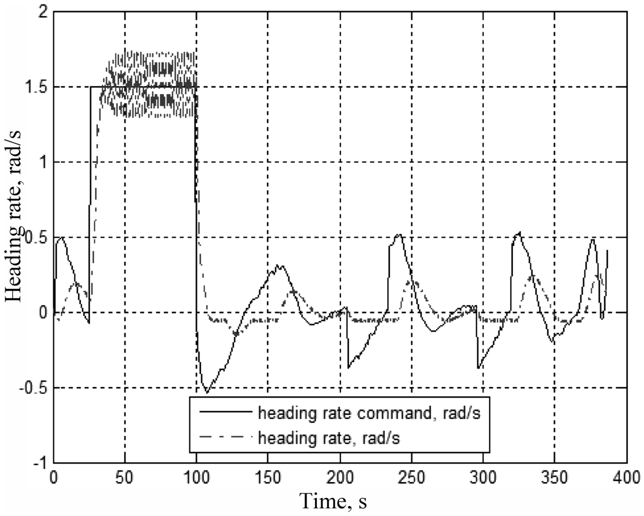

Fig. 17 Tracking performance without adaptation.

Conclusions

There is a need in precision airdrops to design intelligent, self-learning systems that enable users to adopt and deploy equipment in the most practical manner. We have presented one such approach to fulfilling this requirement. The approach to modeling affords a designer easy access to a simulation that can readily be tuned using flight-test data. We have found the approach to be efficient for the purposes of guidance law and stability augmentation design. Among the various flocking strategies investigated, we found the approach of steering toward the air unit closest to the target site to be the best and simplest strategy. The SAS design is sensitive to uncertainty in the servo characteristics, which depends on wing loading, payload shape, and variances that can occur in parafoil construction. Adding adaptation reduces this sensitivity and permits a higher bandwidth design. The overall performance of the adaptive SAS design when operating in both the linear and nonlinear region of the servo characteristic has been demonstrated in simulation to be far superior. Ultimately, the bandwidths of both the adaptive and the nonadaptive SAS design are limited by the presence of oscillations in the suspension system, which suggests that the design can be improved by feedback of bank angle rather than heading rate in the SAS design.

Acknowledgments

This research has been sponsored by the U.S. Army under Phase I and Phase II SBIR efforts. The authors would particularly like to thank Jose Miletti, Richard Benney, and Ed Doucette of the U.S. Army Natick Soldier Center, as well as Ganter Ludwig of Atair Aerospace, and Mark Borgerson and Eric Corban, consultants, for their assistance in the design of and preparation for the flight tests described in this paper.

References

[1] Das, A. V., Fierro, R., Kumar, V., Ostrowski, J. P., Spletzer J., and Taylor, C. J., "A Vision-Based Formation Control Framework," *IEEE Transactions on Robotics and Automation*, Vol. 18, No. 5, Oct. 2002, pp. 813–825.
doi:10.1109/TRA.2002.803463

[2] Sattigeri, R., Calise, A. J., and Evers, J., "An Adaptive Approach to Vision-Based Formation Control," AIAA Paper 2003-5727, Aug. 2003.

[3] Beard, R. W., Lawton J., and Hadeagh, F. Y., "A Feedback Architecture for Formation Control," *Proceedings of the American Control Conference*, Kluwer Academic, Norwell, MA, June 2000, pp. 2087–2091.

[4] Leonard, N. E., and Fiorelli, E., "Virtual Leaders, Artificial Potentials and Coordinated Control of Groups," *IEEE Conference on Decision and Control*, Inst. of Electrical and Electronics Engineers, New York, Dec. 2001, pp. 2968–2973.

[5] Balch, T., and Arkin, R. C., "Behavior-Based Formation Control for Multi-Robot Teams," *IEEE Transactions on Robotics and Automation*, Vol. 14, Nov. 1998, pp. 926–934. doi:10.1109/70.736776

[6] Mataric, M., *Interaction and Intelligent Behavior*, Ph.D. Thesis, Dept. of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, 1994.

[7] Bonabeau, E., Dorigo, M., and Theraulaz, G., *Swarm Intelligence: From Natural to Artificial Systems*, Oxford Univ. Press, England, U.K., 1999, Chap 7.

[8] Reynolds, C. W., "Flocks, Herds and Schools: ADistributed Behavioral Model," *Computer Graphics*, Vol. 21, No. 4, 1987, pp. 25–34. doi:10.1145/37402.37406

[9] Tanner, H. G., Jadbabaie, A., and Pappas, G. J., "Stable Flocking of Mobile Agents, Part I: Fixed Topology," *IEEE Conference on Decision and Control*, Inst. of Electrical and Electronics Engineers, New York, Dec. 2003, pp. 2010–2015.

[10] Olfati, S., and Murray, R. M., "Flocking with Obstacle Avoidance: Cooperation with Limited Communication in Mobile Networks," *Proceedings of the IEEE Conference on Decision and Control*, Inst. of Electrical and Electronics Engineers, New York, Dec. 2003pp. 2022–2028.

[11] Calise, A. J., Yang, B.-J., and Craig, J. I., "Augmenting Adaptive Approach to Control of Flexible Systems," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 3, 2004, pp. 387–396. doi:10.2514/1.2441

[12] Jann, T., "Advanced Features for Autonomous Parafoil Guidance, Navigation and Control," AIAA Paper 2005-1642, May 2005.

[13] Carter, D. W., George, S., Hattis, P. D., McConley, M. W., Rasmussen, S. A., and Singh, L., "Autonomous Large Parafoil Guidance, Navigation, and Control System Design Status," AIAA Paper 2007-2514, May 2007.

[14] Johnson, E. N., and Calise, A., "Limited Authority Adaptive Flight Control for Reusable Launch Vehicles," *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 6, 2003, pp. 926–913.

[15] Zhu, Y., Moreau, M., Accorsi, M., and Leonard, J., "Computer Simulation of Parafoil Dynamics," AIAA Paper 2001-2005, May 2001.

[16] Madsen, C. M., and Cerimele, C. J., "Flight Performance, Aerodynamics, and Simulation Development for the X-38 Parafoil Test Program," AIAA Paper 2003-2108, May 2003.

[17] Muller, S., Wagner, O., and Sachs, G., "A High-Fidelity Nonlinear Multibody Simulation Model for Parafoil Systems," AIAA Paper 2003-2120, May 2003.

[18] Prakash, O., and Ananthkrishnan, N., "Modeling and Simulation of 9-DOF Parafoil-Payload System Flight Dynamics," AIAA Paper 2006-6130, Aug. 2006.

[19] Jann, T., "Aerodynamic Model Identification and GNC Design for the Parafoil-Load System ALEX," AIAA Paper 2001-2015, May 2001.

[20] Rogers, R. M., "Aerodynamic Parameter Estimation for Controlled Parachutes," AIAA Paper 2002-4708, Aug. 2002.

[21] Yakimenko, O. A., and Statnikov, R. B., "Multicriteria Parametrical Identification of the Parafoil-Load Delivery System," AIAA Paper 2005-1664, May 2005.

[22] Calise, A., and Preston, D., "Design of a Stability Augmentation System for Airdrop of Autonomous Guided Parafoils," AIAA Paper 2006-6776, Aug. 2006.

[23] Pomet, J., and Praly, L., "Adaptive Nonlinear Regulation: Estimation from the Lyapunov Equation," *IEEE Transactions on Automatic Control*, Vol. 37, No. 6, 1992, pp. 729–740. doi:10.1109/9.256328