68009

p. 20

In-Flight Demonstration of a Real-Time Flush Airdata Sensing (RT–FADS) System

Stephen A. Whitmore, Roy J. Davis
and John Michael Fife

October 1995

In-Flight Demonstration of a Real-Time Flush Airdata Sensing (RT–FADS) System

Stephen A. Whitmore, Roy J. Davis and John Michael Fife
NASA Dryden Flight Research Center
Edwards, California

1995

ABSTRACT

A prototype real-time flush airdata sensing (RT–FADS) system has been developed and flight tested at the NASA Dryden Flight Research Center. This system uses a matrix of pressure orifices on the vehicle nose to estimate airdata parameters in real time using nonlinear regression. The algorithm is robust to sensor failures and noise in the measured pressures. The RT–FADS system has been calibrated using inertial trajectory measurements that were bootstrapped for atmospheric conditions using meteorological data. Mach numbers as high as 1.6 and angles of attack greater than 45° have been tested. The system performance has been evaluated by comparing the RT–FADS to the ship system airdata computer measurements to give a quantitative evaluation relative to an accepted measurement standard. Nominal agreements of approximately 0.003 in Mach number and 0.20° in angle of attack and angle of sideslip have been achieved.

NOMENCLATURE

ADC	airdata computer
ARTS	Airborne Research Test System
EMI	electromagnetic interference
F[...]	functional that relates states to observations
HI–FADS	high-angle-of-attack flush airdata sensing
Hold_0	number of computational cycles held before initiation of algorithm reset
Hold_1	number of computational cycles held before initiation of parity weighting
Hold_2	number of computational cycles held before initiation of parity reset
j	iteration index
M_∞	free-stream Mach number
NVRAM	nonvolatile random access memory
n	port index
P_n	pressure value, n^{th} FADS port, lbf/ft^2
P_∞	static pressure, lbf/ft^2
q_c	dynamic pressure, lbf/ft^2
RT–FADS	real-time flush airdata sensing
SC	system controller
SRA	Systems Research Aircraft
v	element of measurement noise vector
α	angle of attack, deg
β	angle of sideslip, deg
$\Delta\alpha$	angle-of-attack upwash calibration parameter
$\Delta\beta$	angle-of-sideslip sidewash calibration parameter
δ	perturbation to state or observation vector, 3-σ parameter residual
δP	RT–FADS pressure residual
δP_∞	perturbation to static pressure state estimate, lbf/ft^2
δq_c	perturbation to dynamic pressure state estimate, lbf/ft^2
$\delta\alpha$	perturbation to angle-of-attack state estimate, deg
$\delta\beta$	perturbation to angle-of-sideslip state estimate, deg
ε	static source error calibration parameter
ε_M	variation of ε with Mach number
ε_α	variation of ε with angle of attack
ε_β	variation of ε with angle of sideslip
θ	local flow incidence angle, deg
λ	clock angle of RT–FADS port, deg
σ	standard deviation of parameter residual
ϕ	cone angle of RT–FADS port, deg
χ^2	normalized sum of squared residuals
3-σ	three times the standard deviation of parameter residual

INTRODUCTION

Airdata have always been critical parameters for the flight test community. Conventionally, airdata measurements are performed using intrusive booms that extend beyond the local flow field. These booms measure airmass velocities and incidence angles by direct stagnation of the flow using a pitot tube[1] and mechanical vanes. Although excellent at making steady measurements at low-to-moderate angles of attack, booms are sensitive to vibration and alignment error and are susceptible to damage in flight and on the ground. Airdata measurements have typically been used for research analyses and gain scheduling for flight control systems, but modern multivariable flight control architectures are requiring that airdata be used more extensively for direct flight control or outer-loop guidance.

Specialized requirements of advanced vehicles make use of conventional intrusive airdata measurement systems

highly undesirable. For example, on the X-31 Enhanced Maneuverability Fighter aircraft, the presence of the airdata noseboom caused unsteadiness in the primary forebody vortex cores and induced lateral instabilities at high angles of attack. These instabilities resulted in degraded aircraft handling qualities and, in the worst cases, induced aircraft departures.[2]

In other applications, such as hypersonic aircraft, the hostility of the hypersonic environment mandates the use of nonintrusive airdata systems. On stealth vehicles where a minimal radar cross section is required, such as the B-2 bomber or the F-22 fighter, conventional intrusive systems are highly visible. Eliminating these systems from the basic vehicle design is desirable.

The flush airdata sensing (FADS) system concept was developed as a means of circumventing many of the aforementioned difficulties with intrusive airdata systems. Using this concept, airdata are inferred from nonintrusive surface pressure measurements. The original system prototype was developed for the X-15 program and used a hemispherical nose that was actively steered into the local relative wind vector to measure stagnation pressure and flow incidence angles.[3] The mechanical design of this system was extremely complicated, and the steered-nose concept was abandoned after the X-15 program ended.

A more modern approach, the shuttle entry airdata system, was developed at the NASA Langley Research Center for the space shuttle program.[4] This approach used a matrix of fixed static-pressure measurements and no mechanical actuation of the nose was required. The shuttle entry airdata system technique later was adapted to aeronautical applications, and several demonstration programs were performed in the early 1980s at the NASA Dryden Flight Research Center.[5,6]

For these early programs, measurement and presentation of *individual* pressure coefficient data and their empirical relationships to the airdata parameters were emphasized. These tests verified the feasibility of the fixed-orifice concept but did not attempt to derive algorithms for estimating the airdata from the pressure measurements.

A more advanced program, the high-angle-of-attack flush airdata sensing (HI-FADS) system, was developed and has recently concluded flight testing at NASA Dryden.[7,8] The system was required for the F-18 High Alpha Research Vehicle flight tests because the noseboom installation altered the flow characteristics on the aircraft nose at high incidence angles and adversely affected the vehicle flight dynamics. The HI-FADS design, as with the earlier fixed-orifice FADS systems, used a matrix of flush static-pressure orifices arranged on the nose of the vehicle. This design, however, incorporated the pressure measurements into an overdetermined estimation algorithm where *all* surface pressure observations were used simultaneously to infer the airdata parameters using nonlinear regression.

For the High Alpha Research Vehicle flight tests, the HI-FADS computations were performed postflight using pressure data telemetered to the ground. To allow autonomous operation as part of an actual flight system, the HI-FADS algorithm was integrated into a real-time system that included pressure sensors, computational hardware, onboard program data storage, and an interface to the aircraft instrumentation system. This system, the real-time flush airdata sensing (RT-FADS) system, was flight tested on the NASA Dryden F-18 Systems Research Aircraft (SRA). This paper describes the RT-FADS measurement system, including the basic measurement hardware, the airdata parameter estimation algorithm, and redundancy management schemes that ensure algorithm tolerance to sensor failures. System calibration methods and evaluations of the system performance for subsonic, high-angle-of-attack, and supersonic flight regimes are presented.

THE F-18 SYSTEMS RESEARCH AIRCRAFT

The F-18 SRA is a flight test bed intended for commercial and military systems development and evaluation. The SRA is an F/A-18B aircraft featuring two-place pilot seating, dual engines, and a midwing with leading- and trailing-edge flaps.

A primary objective of the SRA test bed is to identify and flight-test high-leverage technologies beneficial to subsonic, supersonic, hypersonic, or space applications. The SRA flight test bed enables government and industry to focus on the integration, ground test, and flight validation of these technologies.

THE REAL-TIME FLUSH AIRDATA SENSING SYSTEM HARDWARE

Figure 1 shows a basic system overview. The various hardware components, including the flight test radome, the RT-FADS pressure port matrix, and the measurement transducers, are described in the following subsections. The Airborne Research Test System (ARTS) computer used to perform the data logging, onboard computations, and interface to the research instrumentation system is also described.

Radome Configuration

The transducers and the electrical interface unit are mounted on a palette in the SRA radome. The radome for the RT-FADS system is a preproduction sheet metal unit with the flight test noseboom removed that is canted downward at
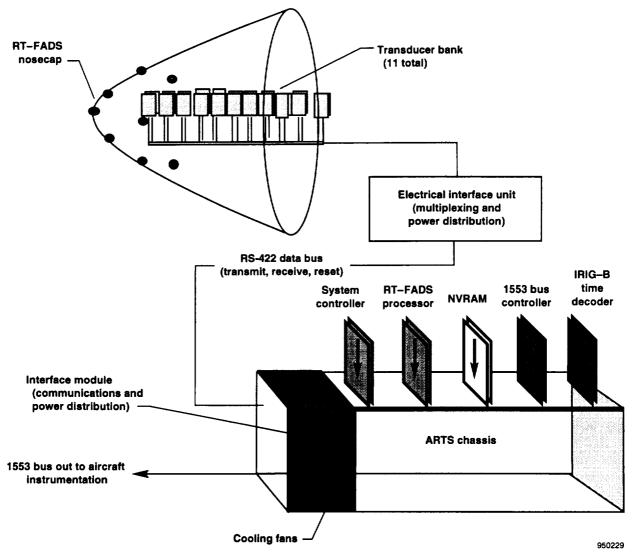
2

RT–FADS nosecap

Transducer bank
(11 total)

Electrical interface unit
(multiplexing and
power distribution)

RS-422 data bus
(transmit, receive, reset)

System
controller

RT–FADS
processor

NVRAM

1553 bus
controller

IRIG–B
time
decoder

Interface module
(communications and
power distribution)

ARTS chassis

1553 bus out to aircraft
instrumentation

Cooling fans

950229

Figure 1. RT–FADS/ARTS system hardware schematic.

an angle of 5.6° relative to the longitudinal axis of the aircraft. A composite matrix nosecap with 11 pressure orifices molded into the structure was attached at the radome tip and faired flush to the surface. The locations of the nosecap ports are defined using cone and clock coordinate angles, ϕ and λ, respectively. Figure 2 shows the clock and cone angles and the pressure port locations on the radome.

The Airborne Research Test System

For the RT–FADS flight tests, the fire-control radar system was removed from the vehicle and the ARTS computer was mounted on a shelf built into the radar mounting rack behind the RT–FADS radome. The ARTS, designed and built at NASA Dryden, is an advanced computer architecture incorporating a VME computer chassis that has been ruggedized for high-stress flight environments. The ARTS architecture

links processor boards with multiple functionalities through a backplane using a commercially available real-time operating system and allows simultaneous multiprocessor, multifunction capability.

Data collection and algorithm computations are performed by two commercially available Motorola 68040–based (Motorola Inc. Microprocessor and Memory Technologies Group; Austin, Texas) single-board computers inserted in the flight-ruggedized chassis (fig. 1). The two processor boards act as slave and master. The master processor acts as the system controller (SC) and manages the data flow through the system. The SC services the slave RT–FADS processor, which in turn communicates to measurement transducers and performs the airdata calculations. Outputs from the RT-FADS processor are passed to the master processor through the ARTS backplane. The SC processor
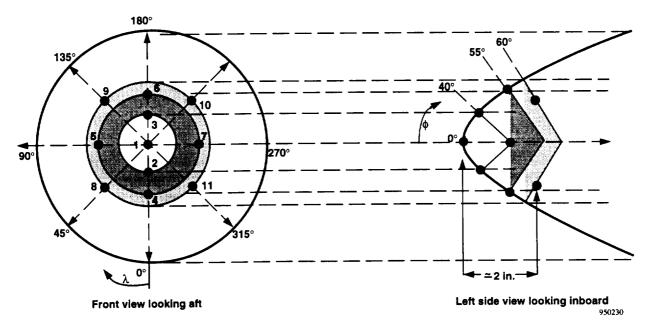
3

Figure 2. Schematic of port locations on the RT–FADS nosecap.

communicates with the aircraft data system through a 1553 bus controller card and a dedicated bus to the instrumentation system.

Other cards in the system include a nonvolatile random access memory (NVRAM) for program and data storage and an IRIG-B time decoder for data time tags. The system has slots to allow for expansion to 8 individual computer boards. An industry-standard high-speed interface unit built into the ARTS chassis allows code updates to be loaded in directly from a remotely located workstation. Manual reset switches in the cockpit allow the pilot to turn off power to the system and reboot the central processing units on demand while in flight. With the exception of the interfaces to the real-time operating system, all code modules were written in the "C" programming language.

Measurement Transducers

Pressures at the FADS ports are sensed by 11 miniaturized, digital absolute-pressure transducers. Each absolute-pressure transducer incorporates a 4-arm active strain-gage bridge for data sensing; internal signal conditioning with a 20 Hz anti-aliasing low-pass filter and output signal amplification; and a 20-bit analog-to-digital conversion. The transducers have a repeatability that exceeds ±0.01 percent of full scale and have a measurement range from 1.50 to 40.00 psia. The digital transducers use nonstandard asychronous serial communication logically similar to the industry-standard RS-485/422/232 protocol.[9] This protocol allows all

of the transducers to be connected through a single common-data bus.

Specifically, the pressure transducers have the capability of serial communication at two Bd rates: a low-speed rate of 9600 Bd, and a special high-speed rate of 375,000 Bd. Because the 9600-Bd rate was considered too slow for the required throughput of the system, the 375,000-Bd capability was exploited in the application. With this high-speed serial-Bd rate, polling a single transducer takes only 0.35 ms. All 11 transducers can be polled in less than 4 ms, allowing sufficient time for the FADS algorithm to achieve the target throughput rate of 50 samples/sec required for flight control feedback. The standard system serial driver on the RT–FADS processor computer board was adapted to support the high-speed serial-Bd rate by reprogramming the RT–FADS processor to use an external clock for data bit timing instead of the standard board system clock.

THE REAL-TIME FLUSH AIRDATA SENSING COMPUTATIONAL ALGORITHM

This section presents the computational algorithm. Figure 3 shows data flow through the RT–FADS algorithm. The FADS/ARTS interface is described. The aerodynamics model that relates the measured pressure data to the airdata parameters is presented. The nonlinear regression algorithm used to solve the system equations and the stability of the
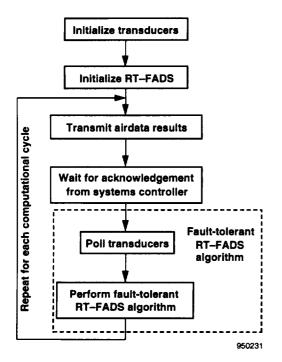
4

```
┌─────────────────────────┐
│  Initialize transducers │
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│   Initialize RT–FADS    │
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│  Transmit airdata results│
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│  Wait for acknowledgement│
│  from systems controller │
└─────────────────────────┘
```

Figure 3. Overview of RT–FADS executive algorithm structure.

regression algorithm is discussed. Finally, algorithm startup and redundancy management routines are developed.

Flush Airdata Sensing/Airborne Research Test System Interface

As previously mentioned, the RT–FADS algorithm is loaded into the modified RT–FADS slave processor from the NVRAM and communicates with the SC processor through the ARTS backplane. When the algorithm is booted, the first function performed is the transducer reset. This function tells the transducers to read and load their hard-wired addresses into memory and to prepare to start delivering data. Next, the transducers are polled sequentially until an entire initial pressure data set is received, and the startup condition is estimated.

When the algorithm has successfully been initialized, the RT-FADS processing loop is entered and computational cycles are performed. Each computational cycle is performed as rapidly as transducer communications and airdata computations allow. At the end of each computational cycle, the RT–FADS processor sets a flag on the ARTS backplane and waits for acknowledgment from the SC processor. Acknowledgment instructs the algorithm to proceed with a new set of computations; the transducers are polled again; and a new set of computations are performed.

The SC processor performs time synchronization with the data rate currently variable from 25 to 100 samples/sec. FADS throughput rates as high as 50 samples/sec were

achieved. If the FADS has not completed a computation cycle when the SC polls the FADS for new data, the SC holds the last values from the previous RT–FADS computational cycles and flags the processing error.

Aerodynamics Model

A pressure model is used to relate the pressure measurements to airdata quantities. The model prescribes measured pressure in terms of four airdata parameters: dynamic pressure, q_c; angle of attack, α; angle of sideslip, β; and static pressure, P_∞. Using these four basic parameters, most other airdata quantities of interest may be directly calculated. As previously reported,[4] the measured pressure data at the nth pressure port is related to the desired airdata parameters by the simple model

$$P(\phi_n, \lambda_n) = q_c[\cos^2(\theta_n) + \varepsilon \sin^2(\theta_n)] + P_\infty \qquad (1)$$

In equation (1), θ_n is the flow incidence angle between the surface at the nth port and the velocity vector. ε is an aerodynamic calibration parameter that must be empirically determined and represents the "static source error" of the system caused by the presence of the aircraft afterbody and the supersonic bow shock wave. This static source error calibration parameter is analogous to the classic "position error" calibration that must be identified for probe-based systems. The incidence angle is related to the local angle of attack and angle of sideslip by

$$\begin{aligned}
\cos(\theta_n) = & \cos(\alpha)\cos(\beta)\cos(\lambda_n) + \\
& \sin(\beta)\sin(\phi_n)\sin(\lambda_n) + \\
& \sin(\alpha)\cos(\beta)\cos(\phi_n)\sin(\lambda_n)
\end{aligned} \qquad (2)$$

The parameters α and β are local flow angles whose values are influenced by the aircraft-induced wash. Their relationships to the free-stream values must also be calibrated. Values for the calibration parameters are presented in the "Flight Test Results" section.

Nonlinear Regression Algorithm

Because the aerodynamic model is nonlinear and cannot be directly inverted to allow calculation of airdata as a function of the measured pressures, the measurements must be used to indirectly infer the airdata state using a nonlinear least-squares regression. Within each computational cycle, the algorithm is linearized about a starting airdata value for each port location as follows:

$$\delta P(\theta_n) = P(\theta_n) - F(\alpha_j, \beta_j, q_{c_j}, P_{\infty_j}, \varepsilon, \phi_n, \lambda_n) \qquad (3)$$

$$\approx \frac{\partial F}{\partial \alpha_j}\delta\alpha + \frac{\partial F}{\partial \beta_j}\delta\beta + \frac{\partial F}{\partial q_{c_j}}\delta q_c + \frac{\partial F}{\partial P_{\infty_j}}\delta P_\infty + noise$$

5

where α_j, β_j, q_{c_j}, and P_{∞_j} are the values about which the algorithm is linearized. Applying equation (3) for all 11 pressure observations, the system may be written as a system of the form:

$$
\begin{bmatrix} \delta P(\theta_1) \\ \cdots \\ \delta P(\theta_{11}) \end{bmatrix} = \begin{bmatrix} \dfrac{\partial F_1}{\partial \alpha} \dfrac{\partial F_1}{\partial \beta} \dfrac{\partial F_1}{\partial q_c} \dfrac{\partial F_1}{\partial P_\infty} \\ \cdots\cdots\cdots \\ \dfrac{\partial F_{11}}{\partial \alpha} \dfrac{\partial F_{11}}{\partial \beta} \dfrac{\partial F_{11}}{\partial q_c} \dfrac{\partial F_{11}}{\partial P_\infty} \end{bmatrix}_j \begin{bmatrix} \delta\alpha \\ \delta\beta \\ \delta q_c \\ \delta P_\infty \end{bmatrix}_j + \begin{bmatrix} v_1 \\ \cdots \\ v_{11} \end{bmatrix} \quad (4)
$$

This overdetermined system of perturbation equations is solved using the weighted least-squares technique.[10] As discussed later in the redundancy management module description, the weighted least-squares technique allows failed or erroneous sensor readings to be eliminated from the estimation algorithm.

At the end of least-squares regression, the resulting perturbation is added to the starting value and the system is relinearized about the resulting update. The iteration is repeated until algorithm convergence is reached — typically in 2 cycles, but as many as 8 cycles. The specific criterion used to determine algorithm convergence is discussed later in the redundancy management module description. At the beginning of each new computational cycle, the system of equations is relinearized about the result of the previous cycle and the iteration is repeated using new pressure data. Development and description of the regression algorithm have previously been reported.[7,8]

Algorithm Startup Module

Because the algorithm computations are recursive — that is, the computations of each computational cycle rely on the results of the previous computational cycle — a reliable means of starting the algorithm computations must be used. The startup algorithm for the RT–FADS is extremely simple but very effective. When the startup algorithm is triggered, the transducers are polled and an initial pressure data set is obtained. This data set is the target pressure set. Starting at a known set of airdata conditions corresponding to a nominal cruise flight condition,* the corresponding pressure set is evaluated using the aerodynamic model. The predicted data set is then linearly perturbed toward the measured data set by some small amount (currently set at 1 percent of the distance between the target set and the starting pressure set).

Using the perturbed value, a new estimate of the airdata state is evaluated using a single cycle of the RT–FADS

---

* These data are user-selectable with a default of Mach 0.50, an altitude of 20,000 ft, and 0° true α and β.

regression algorithm. A new perturbed pressure prediction is evaluated and the iteration is repeated again until the maximum number of iterations (100) have been performed. At that point, the startup algorithm is exited and the RT–FADS computational cycle is begun using the resulting airdata estimate to perform the initial linearization. The startup algorithm is called each time power to the ARTS is cycled or when a manual reset is performed. The startup algorithm typically takes approximately 2 sec to complete.

Algorithm Convergence Issues

In general, the RT–FADS algorithm converges rapidly; however, two potential problems with algorithm convergence have been encountered. The first convergence instability happened under supersonic flight conditions when mixed subsonic/supersonic flow occurred on the RT–FADS pressure matrix. For this supersonic convergence instability, the algorithm did not actually diverge, but the accuracy of the algorithm computations degraded.

The second convergence problem was caused by the failure of one or more pressure measurements. This instability was catastrophic and resulted in algorithm divergence and complete system failure.

Supersonic Instability

From inspection of RT–FADS flight data, the supersonic convergence instability was determined to be caused by location of the sonic line behind the detached shock at the aircraft nose. Figure 4 shows that at low supersonic Mach numbers (less than Mach 1.10), the subsonic region behind the detached shock is large enough to include all of the RT–FADS pressure ports. But as the Mach number increases (greater than Mach 1.25), the sonic line creeps forward so that for certain critical Mach numbers, some ports lie in a subsonic flow regime while some lie in a supersonic flow regime.

When the RT–FADS ports lie in the subsonic/supersonic flow arrangement, then a "multiple minima" solution exists. These minima appear to lie close together. The small pressure perturbations induced by passage of the sonic line are large enough to cause the algorithm to converge to a false minimum. The estimate of static pressure triggers the supersonic instability.

The algorithm was stabilized by low-pass–filtering the static pressure estimate at the end of each computational cycle and using the filtered estimate for linearization in the next computational cycle. Assuming that the solution for the previous computational cycle achieves the true minimum, the low-pass filter tends to keep solution in the proper
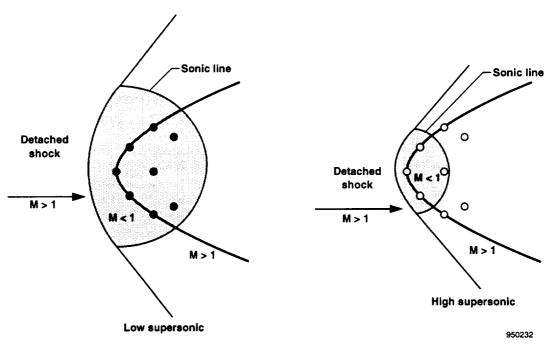
6

Figure 4. Subsonic zone behind detached shock on RT–FADS nosecone.

minima region by not allowing drastic changes from one computational cycle to the next.

Sensor Failure Instabilities

Because the RT–FADS algorithm is nonlinear and the solution is based on small perturbations to a current-state estimate, one true minimum (the physical solution) exists and multiple false minima may exist for each airdata grouping. If a large false perturbation is input to the algorithm — as may happen in the event of a sensor measurement failure — then the algorithm can be disturbed so far away from the true minimum that it converges to a false minimum or diverges altogether. In the former case, the algorithm computes erroneous results. In the latter case, the algorithm fails entirely.

If a false minimum has been reached, the algorithm will not reliably return to the true minimum without reinitializing with a new starting condition. Redundancy management techniques to identify and remove this catastrophic instability have been developed.

Sensor Failure Detection and Redundancy Management Using Analysis of the Normalized Sum of Squared Residuals
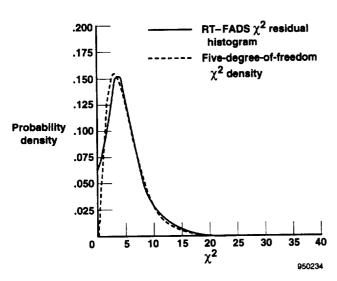
As previously mentioned, when a measurement failure is encountered and a large disturbance occurs, the resulting large false perturbation may cause the algorithm to converge

to a false minimum. To prevent this occurrence, a reliable method for detecting and eliminating failed or invalid measurements from the algorithm is required. At the end of each iteration, the root-mean-square value of the data residuals represents a quantitative measure of the algorithm performance for that iteration. For a converged iteration, the pressure residuals are a subset of a much larger random population whose statistical properties are approximately $0 \ lbf/ft^2$ mean and Gaussian distributed.

Figure 5(a) shows a sample histogram of the residuals normalized by the sample population standard deviation (derived from flight data for converged computations) compared to the normal Gaussian distribution. This histogram validates the assumption that the RT–FADS residuals are normally distributed.

Because the RT–FADS residuals are Gaussian distributed, the sum of the squares of the RT–FADS residuals is a variable whose random distribution is closely approximated by the χ^2 distribution for five degrees of freedom.[11] Five degrees of freedom results from the 11 independent transducer measurements and the fact that the model residuals are related by 6 parameters estimated from the pressure data. These 6 parameters are the sample variance, σ; the calibration parameter, ε; and the four airdata parameters, q_c, P_∞, α, and, β. Figure 5(b) shows a sample sum-square probability distribution of the algorithm residuals for converged RT–FADS flight data compared to a theoretical χ^2 distribution for five degrees of freedom.

(a) Residual distribution.

(b) Normalized residual sum.

Figure 5. Probability distributions for converged RT–FADS algorithm.

Algorithm Convergence and Fault Detection Using the Normalized Sum of Squared Residuals Criterion

At the end of each iteration, comparison of this sample χ^2 variable with percentage points of χ^2 distribution gives a reliable statistical test of whether the algorithm converged. Because χ^2 is a relative probability indicator, a small value of χ^2 corresponding to high probability in the tables indicates that convergence is likely. A large χ^2 value corresponding to a low probability in the tables indicates that convergence is unlikely.

Because of its quantitative indication of convergence probabilities, the χ^2 of the residuals is used as the convergence criterion for the algorithm. When the 90-percent residual confidence level is reached for a particular degree of freedom, then convergence is implicit and one additional iteration is performed.

Redundancy Management and Fault-Tolerant Processing Options

Figure 6 schematically shows the redundancy management schemes employed in the RT–FADS algorithm. Because sensor failures are likely to occur, and because the RT–FADS processing algorithm uses all of the data simultaneously to compute the airdata values, an easy mechanism for removing failed sensors from the algorithm is implemented using the weighted least-squares technique. If a weight is set to zero, then the pressure reading corresponding to the weight has no influence on the estimation.

A series of tests (described in the following paragraph) are performed for each transducer, and the result of each test is either 1 (passed) or 0 (failed). If a transducer reading fails any of the fault processing tests, then it is weighted out of the algorithm and the degrees of freedom of the system are reduced by one. This weighting-out allows up to 4 transducers to be weighted out of the algorithm while still giving an identifiable system (degrees of freedom greater than or equal to 1). The main benefit of the χ^2 methods is they allow a nominal operation that devotes very little of its processing time to data checking. This benefit allows a fast nominal throughput. Only when the χ^2 test fails are the time-consuming data-checking schemes used. This operation is a unique approach to fault-tolerant systems.

When a sensor is polled, timing watchdogs are set to ensure that a response is received within a preset time period. If a correct response has occurred in the preset period, then the measurement weight is set to 1, the watchdog is reset, and the next sensor in the sequence is polled. If a watchdog times out for 100 consecutive computational cycles, then the transducer is considered to have permanently failed and is weighted out of the algorithm. This transducer will no longer be polled until the ARTS power is cycled or a manual reset is performed. When a data reading has been successfully collected, a check is performed to determine that the pressure reading is within a preset reasonableness bound given by

$$100 \ \text{lbf/ft}^2 < \delta P_n < 2250 \ \text{lbf/ft}^2, n = 1...11. \qquad (5)$$

Data values exceeding the reasonableness limits are weighted out of the computation. For each pressure weighted out of the algorithm, the degrees of freedom for the χ^2 test are reduced by 1.
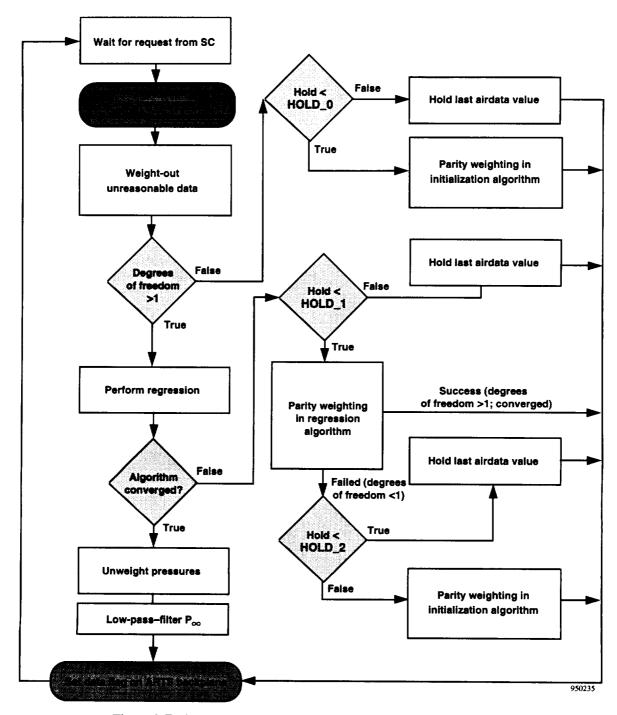
Wait for request from SC

Weight-out unreasonable data

Degrees of freedom >1 — False / True

Perform regression

Algorithm converged? — False / True

Unweight pressures

Low-pass–filter P_∞

Hold < HOLD_0 — False → **Hold last airdata value** / True → **Parity weighting in initialization algorithm**

Hold < HOLD_1 — False → **Hold last airdata value** / True → **Parity weighting in regression algorithm** → Success (degrees of freedom >1; converged)

Failed (degrees of freedom <1)

Hold < HOLD_2 — True → **Hold last airdata value** / False → **Parity weighting in initialization algorithm**

950235

Figure 6. Fault-tolerant data flow through RT–FADS computational cycle.

If more than 4 sensors have failed the watchdog or reasonableness tests, then the system is indeterminate for that computational cycle and a hold-last-value mode is employed. In this mode, the data returned from the algorithm are the last available good airdata values. The hold-last-value mode allows for momentary disturbances such as data spikes to pass without occupying processor time with fault-tolerance computations while still ensuring that the algorithm computations for the next computational cycle will not diverge. The hold-last-value scheme is employed for a maximum of 4 computational cycles.

After all pressures that exceed the reasonableness limits have been removed, the nominal processing loop is entered. After an iteration is performed, the χ^2 variable is recomputed and its value is compared against the table values. If

9

the χ^2 probability is greater than 90 percent in the tables, then convergence is likely and one additional iteration is performed. After the additional iteration, the pressure weights are reset; a computational cycle is written to the ARTS backplane; and the algorithm returns to await acknowledgment from the ARTS so that a new computation cycle can begin.

If the maximum number of allowable iterations (currently 8) have occurred and the degrees of freedom are greater than or equal to 1, the regression cycles are not converging satisfactorily and a hold-last-value on the airdata state using the last available good estimate is again employed. This hold scheme is allowed to occur for up to 2 consecutive iterations. If the criterion has not been violated, then the algorithm outputs data, retains the current weights, and returns to await acknowledgment for a new computational cycle. If the algorithm has not converged for 3 computational cycles, then something is seriously wrong with the data and a more sophisticated scheme to detect the failed pressures is initiated.

A parity weighting scheme is employed in which successive groupings of transducers are weighted out of the algorithm and the regression is performed. The algorithm weights each of the transducers individually, and the regression is performed using the converged airdata from the previous computational cycle as the starting point. The χ^2 variable is computed and is compared against the table values for four degrees of freedom (in each case, the number of transducers has been reduced by 1). If weighting out one of the transducers gives a χ^2 value that has a convergence probability of greater than 50 percent, then the algorithm is iterated until convergence. If convergence occurs, then the weights are reset and the algorithm outputs data and returns to await acknowledgment for a new computational cycle.

If convergence does not occur by weighting out just one of the transducers, then all successive groupings of two are weighted out. The χ^2 variable is computed and compared against the table values for three degrees of freedom (the number of transducers has been reduced by two). If weighting out one of the groupings gives a convergence probability of greater than 50 percent, then the algorithm is iterated until convergence (90 percent). Otherwise, a new grouping is selected. If convergence has not occurred after all possible groupings of two have been exhausted, then groupings of three are evaluated, and so on. When the degrees of freedom of the system drops to less than 1, then the parity weighting scheme has failed; the system is indeterminate; and hold-last-value on the airdata using the last available good estimate is employed.

If the system is still indeterminate after 3 consecutive computational cycles, then the algorithm is considered to have diverged catastrophically and the system is reinitialized

using the startup routine previously described. The watchdog, reasonableness, and parity weighting schemes are also built into the startup routine. Also, at the beginning of any computational cycle, if the watchdog and reasonableness criterion give a degree of freedom less than 1 for more than 4 computational cycles, the data are considered totally unreliable and a system reinitialization is performed.

Fault-Tolerance Summary

Although the fault-tolerance methods sound complex, they are surprisingly simple in their implementation. Only the χ^2 table values corresponding to 90 percent and 50 percent for one through six degrees of freedom are required. Thus, the χ^2 percentage table has only 12 elements. The main feature of these fault-tolerance methods is they allow a nominal operation that devotes very little of its time to data checking. This feature allows a very fast nominal throughput. Only when data failures begin to occur are the more time-consuming data-checking schemes used.

This method is unique to fault-tolerant systems. Furthermore, as new failure situations are encountered, new failure detection criteria can be added without changing the basic structure of the fault management system. Figure 6 shows data flow through the fault-tolerant processing options. More detailed theoretical development of the RT–FADS fault identification and redundancy management schemes have previously been presented.[8]

FLIGHT TEST RESULTS

This section presents flight results for the RT–FADS system. Generally, the integrated RT–FADS/ARTS performed well, with more than 14 flight-hours completed. The system was flight demonstrated from takeoff to landing and throughout the entire nominal flight envelope of the F-18 airplane (to a maximum of 45° α, ± 25° β, and Mach 1.6). The system was cold-soaked at an altitude of 45,000 ft without any noticeable degradation of performance. The algorithm reset and initialization algorithms worked flawlessly on the ground and in flight. The absolute pressure transducers also performed well, and engineering unit calibrations on the individual sensors varied negligibly during the entire 5-month block of Phase I test flights.

Aerodynamic Calibration Procedures and Results

The calibration effort, based on the reference airdata methods,[6] consists primarily of two tasks: identifying the static source error represented by the parameter ε; and identifying the induced wash parameters, $\Delta\alpha$ and $\Delta\beta$. The reference airdata values were generated by merging complementary information from multiple data sources, including the onboard inertial navigation system attitudes, rates, and

accelerations; radar tracking velocity and position data;[12] and rawinsonde weather balloon sounding data.[13]

The weather balloon sounding data were verified inflight by flying 360° at a constant speed, performing a level turn before and after each maneuver. When the indicated airspeed from the RT–FADS system is averaged over the course of the 360° turn, the effects of the winds are eliminated. The difference between the averaged airspeed and the averaged radar-derived groundspeed is the velocity error for that airspeed caused by the static source error. The velocity error provides an accurate point on the position error curve.

The local wind direction and speed are evaluated by adding this static source velocity error to the indicated airspeed reading and then plotting the groundspeed and corrected airspeed as a function of time. Velocity data are converted to Mach number using temperatures values obtained from the rawindsonde balloon soundings and radar-derived geometric altitude. Local ambient pressure values are evaluated using rawindsonde balloon soundings and radar-derived geometric altitude.

The calibration parameters were estimated by substituting the reference airdata into the aerodynamic model and comparing the pressure predictions of the model to the pressures that were actually measured. Residuals between the measured and predicted pressures were then used to infer the values of the calibration parameters at each computational cycle using a nonlinear regression similar to the RT–FADS algorithm. Systematic trends in the calibration parameters were identified by plotting the estimated calibration parameters as a function of flight variables and visually inspecting the results. Once trends were identified by visual inspection, they were curve-fit and interpolated to generate a series of tabular break points that were hard-coded into the RT–FADS algorithm. Figures 7, 8, and 9 show the resulting calibration trends.

Figure 7 shows calibration data for the upwash parameter. The ordinate axis is $\Delta\alpha$, and the abscissa is the local angle of attack measured by the RT–FADS system. The true angle of attack is evaluated by subtracting $\Delta\alpha$ from the indicated value. At moderate subsonic speeds, the upwash correction did not vary with Mach number, but for Mach numbers greater than 0.9, the upwash falls off rapidly as a function of Mach number and an additional correction is needed.

Figure 8 shows similar calibration data for the sidewash parameter, $\Delta\beta$. Because getting significant sideslip angles on the F-18 airplane supersonically is extremely difficult, no supersonic sidewash correction has yet been implemented for the angle of sideslip. Again, the true angle of sideslip is evaluated by subtracting $\Delta\beta$ from the indicated value.
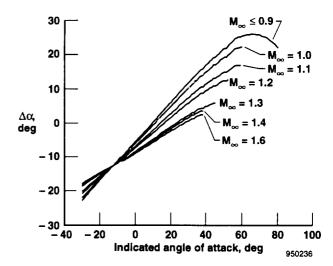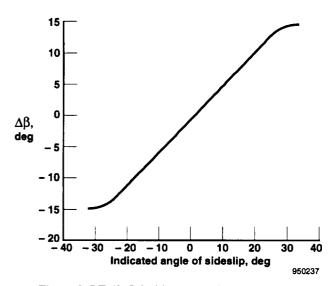

Figure 7. RT–FADS upwash calibration curve.

Figure 8. RT–FADS sidewash calibration curve.

Figure 9 shows a calibration curve for the static source error. It was empirically determined that ε may be decomposed into three distinct components: ε_M, which varies as a function of Mach number; ε_α, which varies as a function of indicated angle of attack; and ε_β, which varies as a function of indicated angle of sideslip. The total static source error is given by the summation

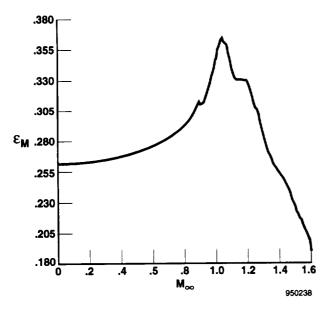$$\varepsilon = \varepsilon_M + \varepsilon_\alpha + \varepsilon_\beta \tag{6}$$

Figure 9(a) shows ε_M plotted on the ordinate axis and Mach number plotted on the abscissa. The curve rises steeply through the transonic flight regime, levels off, and then plunges rapidly at the high supersonic Mach numbers. Figure 9(b) shows ε_α plotted on the ordinate axis and angle of

11

attack plotted on the abscissa. Figure 9(c) shows a similar plot for ε_β. The calibration parameters ε_α and ε_β are adjustments for the loss of total flow energy at the RT–FADS array caused by the increasing average flow angularity over all the pressure ports. These parameters become largely negative for high incidence angles.

At $M_\infty = 0$, the theoretical value of ε for a blunt body in uniform flow[14] is 0.25, which is close to the empirically determined value of approximately 0.262. At large supersonic and hypersonic Mach numbers, the theoretical value of ε predicted by modified Newtonian flow[15] is zero, which is the inclination of ε shown in figure 9(a). Data for pitot-static systems[16] substantiate the trends with incidence angle shown in figures 9(a) and 9(c). Based on these agreements with other theoretically or empirically derived results, the empirically derived trends for ε appear to be physically reasonable.

Evaluation of Code Stability and Fault-Tolerance Methods

During the early phases of the RT–FADS flight tests, electromagnetic interference from the aircraft forward transmitter caused erroneously low readings in several of the RT–FADS pressure sensors whenever the pilot keyed the onboard communications and navigation radio.† Figure 10 (a) shows the corrupted pressure data time histories. This partial failure of the RT–FADS measurement
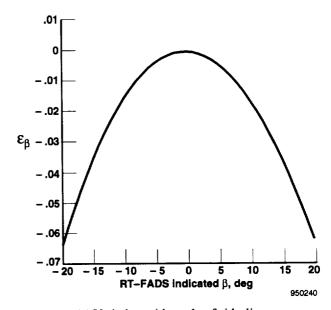

(a) Variation with Mach number.

Figure 9. Static source error calibration data.

---

† The exact cause of this interference was never identified, but the problem was eventually remedied by disconnecting the forward transmitter.

(b) Variation with angle of attack.

(c) Variation with angle of sideslip.

Figure 9. Concluded.

system offered the opportunity to evaluate the performance of the failure detection and fault management techniques.
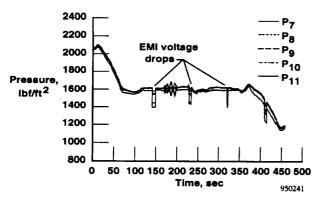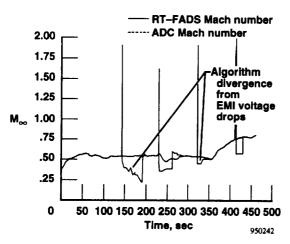
When the algorithm is run off-line with the fault processing options deactivated, the algorithm diverges momentarily, recovers to an erroneous solution, and finally recovers to the correct solution when the pressures return to their correct values. Figure 10(b) shows this failure where the RT–FADS

Mach number estimate is compared to the ship system air-data computer (ADC) value. When the fault detection is activated, the χ^2 values jump rapidly at the instance of measurement failure. The failures are clearly identified and weighted out of the system. Figure 10(c) shows this χ^2 time history. Figure 10(d) shows that when the fault detection mechanisms are activated, only minor fluctuations in the Mach number solution are noted. The fault processing options worked and algorithm divergence was avoided. Other fault-tolerance examples have previously been presented.[7]

Supersonic Instability

Three critical supersonic instability regions were identified for this system. The first critical region lies between Mach 1.25 and Mach 1.30, where the sonic line passes over the furthest aft row of pressure ports. The second critical region lies between Mach 1.45 and Mach 1.50, where the sonic line passes over the next forward row of pressure ports.
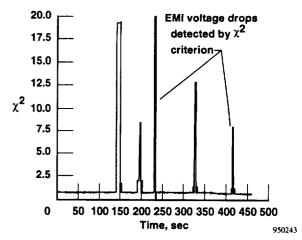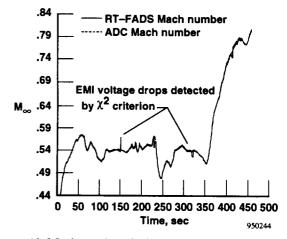

(a) Corrupted pressure data.

(b) Mach number; fault management inactive.

Figure 10. RT–FADS failure detection and fault management example.

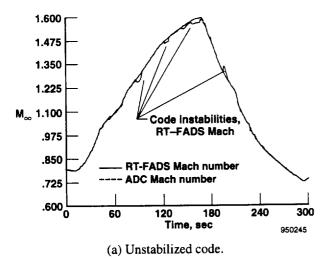(c) χ^2 parameter; fault management active.

(d) Mach number; fault management active

Figure 10. Concluded

The last region of instability lies near Mach 1.60, where the sonic line passes over the most forward row of ports.

Figures 11(a) and 11(b) show the three regions of instability for a Mach 1.6 acceleration/deceleration maneuver. Figure 11(a) shows the time history for the RT–FADS Mach number without the low-pass filter stabilization plotted with the ADC Mach number. The regions of instability are clearly evident by the sudden jumps in the Mach number solution. Figure 11(b) shows the same data, except the low-pass filter on static pressure was used to stabilize the computations.

Figure 12(a) shows the process in more detail for the Mach 1.25 instability. Figure 12(b) shows the time histories of the last row of pressures. A correspondence between the pressure fluctuations and the origin of the instability clearly exist. Note that the stabilized algorithm performs well at the point of instability.
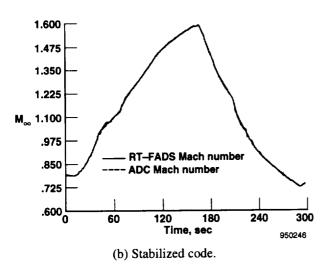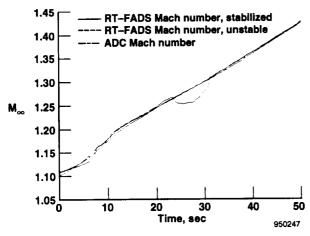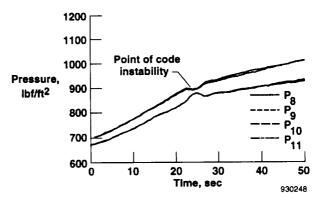
(a) Unstabilized code.

(b) Stabilized code.

Figure 11. Supersonic instabilities in RT–FADS code.

(a) Mach number time history cmparisons.

Figure 12. Detail of Mach 1.25 RT–FADS code instability.

(b) Pressure time histories for last row of RT–FADS pressures (P_8, P_9, P_{10}, and P_{11}).
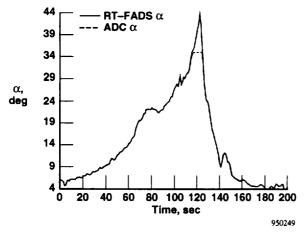
Figure 12. Concluded.
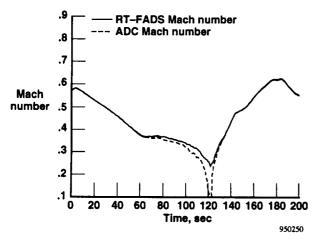
Evaluation of the System Accuracy

Quantitative RT–FADS system performance was evaluated by comparing the calibrated outputs to the ship system ADC outputs. Although the ADC outputs are subject to the same types of measurement errors as the RT–FADS system, the ADC provides an accepted standard for comparison. Figure 13 shows sample time history comparisons for a high-angle-of-attack flight maneuver. Figure 13(a) shows time history comparisons for angle of attack. Figure 13(b) shows Mach number comparisons.

Both angle-of-attack traces are similar to 33° α, the limit of the ADC angle-of-attack range. Because of total pressure loss on the ADC probe pitot port, the ship system Mach number begins to degrade at approximately 25° α. Clearly, the RT–FADS equals the performance of the F-18 ADC at low angles of attack and extends the accurate measurement range to angles of attack higher than 25°. Figure 14 shows similar comparisons for a Mach 1.35 supersonic acceleration/deceleration maneuver.

Assuming that the ADC is the "truth set" and ignoring angles of attack greater than 25° (where the ADC data begins to lose accuracy), the statistical accuracy of the RT–FADS was evaluated as a function of Mach number. This evaluation was performed by taking root-squared residuals between the ADC and RT–FADS measurements and graphing the residuals on a scatter plot as a function of Mach number for all of the data points gathered in the Phase I flight tests — a database of approximately 1,000,000 time computational cycles. Starting at Mach 0.20 and extending to Mach 1.60 at intervals of Mach 0.20, residual boundaries were drawn so that more than 99.9 percent of the residuals in each Mach number interval are included. These residual boundaries establish the 3-σ error bounds for the RT–FADS (relative to the ADC reference) parameters as a function of Mach number.
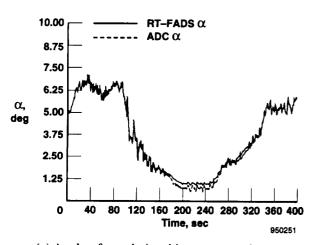
14

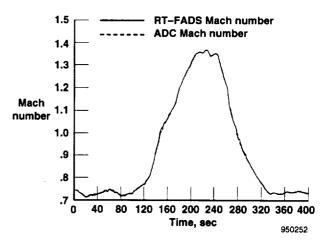(a) Angle-of-attack time history comparison.

(b) Mach-number time history comparison.

Figure 13. Comparison of RT–FADS airdata to airdata computer for high-angle-of-attack maneuver.

(a) Angle-of-attack time history comparison.

Figure 14. Comparison of RT–FADS to airdata computer for Mach 1.35.

(b) Mach-number time history comparison.

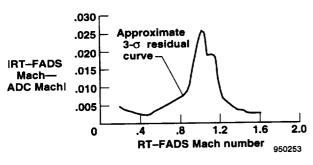Figure 14. Comparison of RT–FADS to airdata computer for Mach 1.35.

Figure 15 shows the residual scatter plots and 3-σ error boundaries for Mach number and angle of attack.

For angles of attack below 25°, the RT–FADS measurement accuracies were nearly independent of angles of attack and sideslip. Because no ship system value was available for angle of sideslip, no statistical evaluation was performed. Because the angles of attack and sideslip were calibrated using the same reference source, inferring that the accuracy levels are very similar is reasonable. Table 1 presents 3-σ residual values as functions of Mach number for various RT-FADS airdata parameters.

Because the ship ADC is believed to have the same general error magnitudes as the RT–FADS, the error values shown in table 1 are conservative. Even if the ADC reference is assumed to be a perfect source, the 3-σ residuals are well within accepted standards for airdata measurement accuracies.

CONCLUDING REMARKS

A novel nonintrusive airdata sensing system was developed and flight tested at the NASA Dryden Flight Research Center. This system uses a matrix of flush pressure orifices arranged on the aircraft forebody to determine the vehicle airdata — critical measurement parameters for flight control and research analyses. The system eliminates the need for external probes that are sensitive to vibration and alignment error, are easily damaged, may alter the flying qualities of the aircraft at high angles of attack, and are unacceptable for hypersonic or stealthy vehicles.
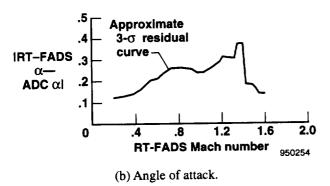
15

(a) Mach number.

(b) Angle of attack.

Figure 15. 3-σ residual scatter plot for RT–FADS system.

The real-time flush airdata sensing (RT–FADS) system is significantly more advanced than earlier nonintrusive airdata concepts. The development of several innovations has resulted in a system that is capable of operating autonomously in real time. The system incorporates an overdetermined algorithm in which all surface pressure observations are used simultaneously to infer the airdata parameters using nonlinear regression. This innovation provides a system that is robust to noise in the measured pressure data and allows multiple sensor losses without significantly degrading the airdata computations. Generally, the integrated system performed well, with more than 14 operational flight hours completed. The system was flight demonstrated from takeoff to landing and over the entire nominal flight envelope of the F-18 airplane (to a maximum of 45° α, ± 25 β, and Mach 1.6).

Flight calibrations were performed using reference airdata values generated using data sources that include the ship system inertial navigation system, radar-tracking velocity and position data, and rawinsonde weather balloon sounding data. The calibration parameters were estimated by substituting the reference airdata into the aerodynamic model and using a nonlinear regression to identify the calibration parameters. Systematic trends in the calibration parameters were identified by plotting the estimated calibration parameters as a function of flight variables and visually inspecting the results. Trends were identified by visual inspection, and curve-fit to generate a series of tabular breakpoints that were hard-coded into the RT–FADS algorithm. Based on

Table 1. 3-σ Residual values for RT–FADS parameters.

M_∞	δM_∞	$\delta\alpha$, deg	δq_c, lbf/ft^2	δP_∞, lbf/ft^2
0.2	0.0045	0.125	3.50	1.00
0.4	0.0025	0.145	2.00	1.00
0.6	0.0035	0.210	2.00	3.25
0.8	0.0070	0.260	2.00	3.50
1.0	0.0240	0.240	12.00	16.50
1.2	0.0100	0.300	6.00	3.00
1.4	0.0030	0.380	5.00	2.00
1.6	0.0025	0.150	3.00	1.00

16

agreements with other theoretically or empirically derived results, the empirically derived trends for the static source error calibration parameter, ε, appear to be physically reasonable.

Two types of algorithm instabilities were identified. The first type of instability occurred during supersonic flight. The source of this instability is the location of the sonic line behind the detached normal shock on the aircraft nose. As the Mach number increases, the sonic line creeps forward so that for certain critical Mach numbers, some ports lie in a subsonic flow regime, while some lie in a supersonic flow regime, resulting in a multiple minima. Small pressure perturbations induced by passage of sonic line are large enough to cause the algorithm to diverge to the false minimum. The RT–FADS algorithm was stabilized reasonably well by low-pass–filtering the static pressure estimate and then using the filtered estimate for linearizing between computational cycles. The low-pass filter tends to keep the solution in the proper minima region by not allowing drastic changes from one computational cycle to the next.

A second type of instability was induced by measurement failures, and a χ^2 fault detection and redundancy management scheme was developed to stabilize the algorithm. These methods worked well for all of the Phase I test flights.

The RT–FADS equals the performance of the F-18 ADC at low angles of attack and extends the accurate measurement range to considerably high angles of attack. The statistical accuracy of the RT–FADS was evaluated by taking root-squared residuals between the ADC and RT–FADS measurements and graphing the residuals as a function of Mach number. Residual boundaries were drawn so that more than 99.9 percent of the residuals in each Mach number interval are included. These residual boundaries established a 3-σ error bound for the RT–FADS parameters as a function of Mach number. Because the ship ADC is believed to have the same general error magnitudes as the RT-FADS, the calculated 3-σ residual boundaries are conservative estimates of the true 3-σ error bounds. Even so, the resulting accuracy estimates are well within accepted standards for airdata measurement accuracies.

REFERENCES

[1] Gracey, William, *Measurement of Aircraft Speed and Altitude*, NASA RP-1046, 1980.

[2] Cobleigh, Brent R. and Del Frate, John, *Water Tunnel Flow Visualization Study of a 4.4% Scale X-31 Forebody*, NASA TM-104276, 1994.

[3] Cary, John P. and Keener, Earl R., *Flight Evaluation of the X-15 Ball-Nose Flow-Direction Sensor as an Air-Data System*, NASA TN-D-2923, 1965.

[4] Siemers, P. M. III, Wolf, H., and Henry, M. W., "Shuttle Entry Air Data System (SEADS)-Flight Verification of an Advanced Air Data System Concept," AIAA-88-2104, May 1988.

[5] Larson, Terry J., Whitmore, Stephen A., Ehernberger, L. J., Johnson, J. Blair, and Siemers, Paul M. III, *Qualitative Evaluation of a Flush Air Data System at Transonic Speeds and High Angles of Attack*, NASA TP-2716, 1987.

[6] Larson, Terry J., Moes, Timothy R., and Siemers, Paul M. III, *Wind-Tunnel Investigation of a Flush Airdata System at Mach Numbers From 0.7 to 1.4*, NASA TM-101697, 1990.

[7] Whitmore, Stephen A., Moes, Timothy R., and Larson, Terry J., *Preliminary Results From a Subsonic High Angle-of-Attack Flush Airdata Sensing (HI–FADS) System: Design, Calibration, and Flight Test Evaluation*, NASA TM-101713, 1990.

[8] Whitmore, Stephen A. and Moes, Timothy R., *Failure Detection and Fault Management Techniques for a Pneumatic High-Angle-of-Attack Flush Airdata Sensing (HI–FADS) System*, NASA TM-4335, 1992.

[9] Putnam, Byron W., *RS-232 Simplified: Everything You Need to Know About Connecting, Interfacing, and Troubleshooting Peripheral Devices*, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1987.

[10] Franklin, Gene F. and Powell, J. David, *Digital Control of Dynamic Systems*, Addison Wesley Publishing Company, Reading, Massachusetts, 1980.

[11] Bendat, Julius S. and Piersol, Allan G., *Random Data: Analysis and Measurement Procedures*, Wiley-Interscience, New York, 1971.

[12] Haering, Edward A., Jr., and Whitmore, Stephen A., *FORTRAN Program for Analyzing Ground-Based Radar Data: Usage and Derivations, Version 6.2*, NASA TP-3430, 1994.

[13] Ehernberger, L. J., Haering, Edward A., Jr., Lockhart, Mary G., and Teets, Edward H., "Atmospheric Analysis for Airdata Calibration on Research Aircraft," AIAA-92-0293, Jan. 1992.

[14] Curie, I. G., *Fundamental Mechanics of Fluids*, McGraw-Hill, Inc., New York, 1974.

[15] Anderson, John D., Jr., *Hypersonic and High Temperature Gas Dynamics*, McGraw-Hill, Inc., New York, 1989.

[16] Richardson, Norman R. and Pearson, Albin O., *Wind-Tunnel Calibrations of a Combined Pitot-Static Tube, Vane-Type Flow-Direction Transmitter, and Stagnation-Temperature Element at Mach Numbers from 0.60 to 2.87*, NASA TN-D-122, 1959.

REPORT DOCUMENTATION PAGE	Form Approved OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE October 1995	3. REPORT TYPE AND DATES COVERED Technical Memorandum	
4. TITLE AND SUBTITLE In-Flight Demonstration of a Real-Time Flush Airdata Sensing (RT–FADS) System			**5. FUNDING NUMBERS** WU 505-68-40
6. AUTHOR(S) Stephen A. Whitmore, Roy J. Davis and John Michael Fife			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) NASA Dryden Flight Research Center P.O. Box 273 Edwards, California 93523-0273			**8. PERFORMING ORGANIZATION REPORT NUMBER** H-2053
9. SPONSORING/MONOTORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, DC 20546-0001			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER** NASA TM-104314

11. SUPPLEMENTARY NOTES

Prepared for the American Institute of Aeronautics and Astronautics Atmospheric Flight Mechanics Conference, Baltimore, Maryland, August 7–10, 1995.

12a. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified—Unlimited Subject Category 06	12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

A prototype real-time flush airdata sensing (RT–FADS) system has been developed and flight tested at the NASA Dryden Flight Research Center. This system uses a matrix of pressure orifices on the vehicle nose to estimate airdata parameters in real time using nonlinear regression. The algorithm is robust to sensor failures and noise in the measured pressures. The RT–FADS system has been calibrated using inertial trajectory measurements that were bootstrapped for atmospheric conditions using meteorological data. Mach numbers as high as 1.6 and angles of attack greater than 45° have been tested. The system performance has been evaluated by comparing the RT–FADS to the ship system airdata computer measurements to give a quantitative evaluation relative to an accepted measurement standard. Nominal agreements of approximately 0.003 in Mach number and 0.20° in angle of attack and angle of sideslip have been achieved.

14. SUBJECT TERMS Airdata; Flight testing; Flush port measurements; F-18 HARV; F-18 Systems Research Aircraft; Instrumentation; Pressure sensing			15. NUMBER OF PAGES 20
			16. PRICE CODE AO3
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT Unlimited

NSN 7540-01-280-5500   Available from the NASA Center for AeroSpace Information, 800 Elkridge Landing Road, Linthicum Heights, MD 21090; (301)621-0390

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102