

Wind Measurements in SOCRATES: An Evaluation of the Pitot-Static Sensor and the Gust Pod

with suggested processing algorithms

Al Cooper

11 February 2018

National Center for Atmospheric Research
Earth Observing Laboratory
Research Aviation Facility
cooperw@ucar.edu

Table of Contents

1	Introduction	1
1.1	General scope of this report	1
1.2	Comments related to the standard radome-based system	2
2	Measurements from LAMS – ARISTO-2017	5
2.1	Data processing	6
2.2	Results	7
2.3	Recommendation: PCor function	8
3	Study of the Pitot-Static Sensor	8
3.1	A proposed algorithm	12
4	Using the Gust Pod in SOCRATES	18
4.1	The general approach	18
4.2	Details of the Transformation	20
4.3	Empirical Coefficients	20
4.4	<i>Summary</i>	29
5	High-Rate Measurements	30
5.1	Constructing the high-rate data.frame	30
5.2	Removing spikes	31
5.3	Lateral variance spectra	31
5.4	Longitudinal variance spectra	37
6	Summary and Conclusions	39
7	The Processor	41
A	Reproducibility	42

Abstract

This document discusses some of the options for measuring wind in SOCRATES. The NSF/NCAR GV has some new capabilities that might improve the measurements of wind, but they are not fully investigated and documented yet and in some cases the processing algorithms have not been developed. This report describes some studies of the new measurements and recommends new procedures and processing algorithms for the calculation of wind. The result is evidence that the new pitot-static sensor provides a much improved measurement of airspeed but still has some residual noise at frequencies above about 5 Hz. The gust pod as processed here provides valid measurements of the wind that are consistent with the standard system and so could serve as back-up measurements to that system. However, the variance spectra from the gust pod have several suspicious traits and should not be used for spectral analysis without further study. An additional result documented here is that the corrections now applied to measurements of ambient and dynamic pressure, developed in 2014, remain valid and do not need revision.

1 Introduction

1.1 General scope of this report

The NCAR Technical Note on “Characterization of Uncertainty in Measurements of Wind from the NSF/NCAR GV” (Cooper et al. [2016]) includes an extensive discussion of the wind-measuring capabilities of the GV, and a journal article (Cooper et al. [2014]) documented how a laser air-motion sensor can be used to improve those measurements. In addition, a recent technical note (Cooper [2017]) presented an approach to improving the measurements through use of a Kalman filter. In most regards, these studies concluded that the wind measurement from the GV has low uncertainty limits and meets most needs of research projects. However, one aspect has remained troublesome: The variance spectra of the measurements, especially of the longitudinal component of the wind, show unacceptable noise. This is documented in the 2016 NCAR Technical Note referenced above, in an appendix that discusses the high-frequency measurements (titled “Turbulence”). Plots in this discussion, taken from a suggestion and analysis by Don Lenschow, indicate that the problem is resonance noise in the long lines used to connect pressure sensors to the relevant ports.

In response, RAF has installed a new pitot-static sensor for the measurement of the airspeed. That instrument uses much shorter lines, so it is hoped that it will reduce the noise. In addition, the gust pod is being flown in SOCRATES as another candidate for making high-frequency measurements, because its lines are still shorter. The gust-pod five-hole probe itself is only about an inch in diameter, so it may provide the best high-frequency measurements. Although the sensor was used before in the DEEPWAVE project and provided good measurements, as documented in the 2016 technical note, it is flown in a new configuration in SOCRATES that includes the large pods under the wings. These were not present in DEEPWAVE and apparently change the distortion of airflow approaching the gust pod to such a degree that the processing developed for DEEPWAVE does not work in SOCRATES.

One additional source of information about the wind-sensing system is the ARISTO-2017 experiment, in which the Laser Air-Motion Sensor (LAMS) was flown on the GV. This provides an opportunity to check and possibly update the airspeed calibration reported in the 2014 paper. The large pods were present in ARISTO-2017, so if there is any effect on the standard system it should be possible to detect it by using these new measurements.

This report is organized into the following major sections:

- Section 2 discusses the ARISTO-2017 flights and the use of LAMS to check the measurement of airspeed.
- Section 3 discusses the new pitot-static sensor and proposes a way to use its measurement for the calculation of wind.
- Section 4 develops a new approach to processing the data from the gust pod and shows results from that instrument.
- Section 5 then examines the high-rate measurements available from the standard system, the standard system modified to use the pitot-static sensor, and the gust-pod system.

The “Summary” section then discusses proposed ways to process the data, the relative merits of the two new options, and the general status of wind measurements a high rate. Suggested processing schemes are summarized there for possible use in SOCRATES. Those processing schemes have been incorporated into the “QAtools” shiny app (see this URL) so that tests can be made during and after the field campaign by adding new processed variables to copies of the netCDF files. With appropriate revision, what is proposed can be incorporated into standard processing (in “nimbus”), but the second-pass processing may be a better way to add these variables in the short term.

The present text document and the data processing are incorporated into a single file, named “WindInSOCRATES.Rnw”, located in the EOL directory `~cooperw/RStudio/SocratesQA/`. A “Reproducibility” Appendix discusses how this work could be duplicated, and all the needed components are archived in a GitHub directory, as discussed in that Appendix.

1.2 Comments related to the standard radome-based system

The approach to the gust-pod will be to match the results from the radome system, so that system is revisited here first. A script produced earlier for ORCAS, `AKRD.Rnw`, implemented second-pass processing for AKRD using a complementary-filter approach and developed appropriate coefficients and formulas. A memo at this URL¹ documents those results, and a presentation made in support of this approach is available at this URL at this URL. Because implementation of that approach is still pending, some of the information is repeated here. In addition, data from early SOCRATES flights are added to the data set used previously to test that the results are appropriate for SOCRATES. (It will be useful to repeat this analysis with additional SOCRATES flights when they become available.)

For reference, the “standard” representation in use for several years has been

$$\alpha = c_0 + \frac{\Delta p \alpha}{q} (c_1 + c_2 M)$$

and the coefficients were found by fitting that formula to a reference that assumes there is zero vertical wind:

$$\alpha^* = \theta - \frac{w_p}{V}$$

The justification for this form is that studies of five-hole pressure sensors have found a Mach-number dependence that affects their sensitivity.

Here the approach will be different. The reference value is split into two components, $\alpha^* = \alpha_f^* + \alpha_s^*$ that result from applying a Butterworth low-pass filter to α^* (in the code, the variable `AOAREF`) to obtain α_s^* and then finding α_f^* from $\alpha_f^* = \alpha^* - \alpha_s^*$. These components are then represented by separate fits:

¹Here and elsewhere in this document, references like “this URL” are live links to the appropriate documents.

$$\alpha_f = \left(\frac{\{\text{ADIFR}\}}{\{\text{QCF}\}} \right)_f (c_1 + c_2 M) \quad (1)$$

$$\alpha_s = d_0 + d_1 \left(\frac{\{\text{ADIFR}\}}{\{\text{QCF}\}} \right)_s + d_2 \{\text{QCF}\}_s \quad (2)$$

where the f and s subscripts represent the high-pass and low-pass components after filtering. More complicated representations were tested in both cases, but these appear to provide adequate fits without additional terms. The c_2 term representing Mach-number dependence does not appear to be necessary, but the expectation that there will be some dependence on M justified its tentative inclusion. However, the fits below show no significant difference with $c_2 = 0$, and an analysis of variance made its inclusion appear questionable, so it has been dropped from the calculations presented here.

This approach has two advantages:

1. The important sensitivity to fluctuations is not compromised by efforts to represent the slowly varying zero level for angle of attack.
2. The slowly varying zero reference can be represented by more complex equations without needing to apply those to the high-frequency component.

The result is an empirical representation that is appropriate for all recent GV projects and that does not need to be changed for each project, as has been the case for many recent projects that use the standard representation. Readers may prefer to jump to Section 2 and skip the details of how this representation was developed because this information has been presented before at the Research Aviation Facility. The rest of this subsection is only a minor update to the previous study that uses some of the measurements from SOCRATES.

Data used in the fit

The selection of data was described in the previous memo, but that information will be repeated here with the addition of SOCRATES data. The first step is to assemble the data to be used for fitting. The data set should include as many flights and projects as is reasonable, but should be reviewed also to eliminate flights that appear anomalous because of problems with the measurements, strong updrafts and downdrafts, special loading conditions, or other aspects of the flights that make them questionable to include.

In this case, measurements from ORCAS, CSET, DEEPWAVE and early SOCRATES were used. For CSET, the selected flights were 1, 2, 3, 4, 6, 10, 12, and 16. For ORCAS, flights 1, 2, 3, 6, 8, 9, 11, 14, and 16 were selected. For DEEPWAVE, all 26 flights were used excepting 6, 7, 15, and 23. For SOCRATES, test flight 1 is particularly useful to include because it includes the only maneuvers to data. Research flights 1 and 2 are the only ones available to data and are needed to emphasize the low-level flight segments that are important conditions for the measurement of wind, but for now they have been excluded because the vertical-wind variance spectra suggest some problem with ADIFR. To use different flights in the future, changes to the

'Project' statements and 'best' statements will be needed, and in addition it will be necessary to change the logical variables 'NewCoefficients' and 'ReloadData' to TRUE. Otherwise, this program skips recalculation of the coefficients and just retrieves them from the file './AKRD-fit-coef.Rdata' where they have been stored by a previous run. The process of constructing the data.frame used in the fit, not necessary if those previously determined coefficients are to be used, was as follows:

1. Read the netCDF file as on /scr/raf_data/{Project}, for each flight in CSET and ORCAS (not just the ones listed above).²
2. For each file, add appropriate variables for fitting ($QR=ADIFR/QCF$, M =Mach number using uncorrected pressures, $AOAREF=PITCH-(GGVSPD/TASF)*180/\pi$, QCF) and the low-pass and high-pass components of these produced by a Butterworth third-order filter with cutoff frequency of 1/600 Hz. A version of the filter was used that made two passes, one forward and one backward, through the data and combined the results. Some exploration of values for the cutoff frequency led to similar results over a wide range from 0.01 Hz to 0.001 Hz, but the choice (1/600) Hz appeared to be a good compromise between the conflicting requirements to represent the low-pass component well without having it distort the high-pass response.
3. The file was truncated to include only data spanning from the first measurement of air-speed above 90 m/s to the last, to avoid periods when the aircraft was still on the ground or just after take-off.
4. A variable representing flight number was added to each file. To avoid ambiguity among projects, ORCAS flights were assigned numbers equal to 100 plus the flight number. This variable, named RF, then made it possible to identify individual flights after all were concatenated.
5. The individual flights were then concatenated into one data.frame containing only the variables needed for fitting and a few others used while examining the results. The variables in the data.frame were ADIFR, AKRD, GGALT, GGVSPD, PITCH, PSF, QCF, ROLL, SSLIP, TASF, TASX, THDG, WIC, and those added in step 2 above.
6. From this data frame, another (called DF) was constructed to use in fitting. It consisted only of the 17 selected flights listed above. In addition, measurements spanning 600 s from the start and end of each file were removed to avoid periods where end-effects seemed to cause problems with the filtered results and also to avoid problematic periods during initial climb and final descent, which otherwise seemed to distort the fits.
7. The final data.frame (DF) was then restricted to measurements with $TASF > 110$ and $ROLL$ between -2 and 2° , to avoid possible periods of slow flight or in turns. Turns in particular invalidate the fit assumption involved in finding the reference (AOAREF) used for the fit, so these need to be excluded or turns cause serious distortion of the results.

²For ORCAS flight 12, the available netCDF file was bad and couldn't be read, so that flight was skipped.

Fit results

The result of fitting using (1) was $c_1 = 19.6373$, but an earlier fit to more selective speed-run and other data in DEEPWAVE suggested a value of $c_1 = 21.481$, as described in the NCAR Technical Note on Wind Uncertainty (Cooper et al. [2016]). That appears to provide a good representation of the CSET/ORCAS/SOCRATES data as well, so that coefficient will be used here. For the low-pass component, the coefficients from the fit to (2) were $\{d\} = \{4.5111, 19.8409, -0.00188\}$, and the residual standard deviation for this fit was 0.06. These results are discussed further in a presentation that is available here, where the choice for c_1 is also explained.

2 Measurements from LAMS – ARISTO-2017

The purpose of this section is to consider if the LAMS measurements from ARISTO2017 are consistent with the previous study of the PCor function as presently documented in the Processing Algorithms Technical Note. During this GV project, the LAMS was flown in a 4-beam configuration. The beam assignments were non-standard, though: Beam 2 was approximately in the forward direction, and beams 1, 3 and 4 were approximately 35° from the longitudinal axis with beam 1 upward, beam 3 outboard and downward, and beam 4 inboard and downward. Relative to an azimuthal angle starting from the downward direction they were, respectively, 180 , 60 and -60° in clockwise rotation about the forward longitudinal axis. Although there were four beams, the fourth seldom produced useful measurements so it will be necessary to use the three-beam solution for the wind vector.

There were six research flights in ARISTO-2017, but the LAMS was not operational on the last two so those will not be used here. Research flight 4 was particularly useful because it extended to low level and so expanded the envelope of measurements that can constrain the pressure-correction algorithms.

For ARISTO-2017, the line-of-sight beam speeds were determined in two ways. Matt Hayman improved upon the processing method originally used by Scott Spuler in the 2014 study. This usually worked very well. Because there was tight coupling to the expected measured TAS (with adjustment for the beam angles), it might be suspect that the method may be biased toward the TAS value at times that have a marginal signal. Therefore, a second method was to fit the background using Savitzgy-Golay polynomials and then search for peaks above this background, as was done in the original 2014 PCor determination. When signals were strong, the method usually produced the same result as the PCA method, but it was more prone to failure during periods where the signal was weak. In this report, Matt's processing will be used with additional restrictions requiring his estimate of uncertainty to meet specified tests for the measurement to be accepted. Because there are many good measurements, this helps avoid cases where the measurements might be suspect and perhaps biased toward the standard TAS.

Although there are many uses of the LAMS measurements, as described in previous reports and memos, here the focus will be on checking the PCor function currently in use and revising it if necessary. The function can be found at this URL, on pages 29–31.

2.1 Data processing

The original netCDF files, with names like ARISTO2017rf01.nc, contain 512-element arrays of the LAMS frequency measurements each time interval, with names like BEAM{1-4}_LAMS. For ARISTO-2017, Matt Hayman produced auxiliary files with names like ARISTO2017rf04_LAMS_TAS_Update.nc. These files contain 28 new variables, in particular line-of-sight speeds (V_LOS_Beam{1-3}) and Cartesian-coordinate velocity components (AIRSPEED_{X,Y,Z}_LAMS) in the LAMS coordinate system. The files also include variables V_LOS_Beamx{1-3}_Uncertainty, estimates of the uncertainty in the line-of-sight airspeeds, which have values ranging from 0 to 10.³ Additional variables (AIRSPEED_{X,Y,Z}_EST) are the result of Kalman-filter adjustment using measurements from LAMS and the standard system, so they should not be used for the present study. Some additional variables are present in these files, but the line-of-sight speeds will be the primary input used here.

The processing chain used for this report was as follows:

1. A Python routine, with minor modifications from that used previously in the 2014 study and the C-130 study that used ARISTO-2015 and ARISTO-2016, was used to find line-of-sight airspeeds along all the beams. It is LAMS_ARISTO.py and the version used is in the directory ~cooperw/RStudio/ARISTO-2017. It reads the original netCDF file containing the beam histograms in variables BEAMx_LAMS, each a 512-element histogram.⁴ The new variables produced by this Python routine are BEAMxspeed where x is {1-4}.
2. The variables V_LOS_Beam{1,2,3} are produced by Matt's processing. They are in separate netCDF files that have only the new variables, so they must be merged into the composite data.frame so they can be used conveniently with the other standard variables and the BEAMxspeed variables. This is done by the R code in the "merge: chunk, which is included here for reference and is not executed because all four beams have been processed in this way and saved in a composite data.frame named DataM.⁵ The individual flights are identified by the variable RF containing the flight number, so individual flights can be selected via, e.g., DataM[DataM\$RF=4,].
3. The subsequent analysis is then contained in this file (WindInSOCRAATES.Rnw). Preliminary checks indicated good agreement between Matt's processing and the Savitzky-Golay approach, so if this study is repeated in the future it will be preferable to omit the Python routine and use only Matt's variables.

³It appears that values of 0 are where there is no measurement (e.g., before takeoff), values of 10 indicate low-uncertainty estimates that perhaps should be excluded from the present study, and the remaining values mostly range between 0 and 1 and are the values to be used in this study. Most are below 0.1, so it may be useful to consider only those measurements with associated uncertainty larger than 0 but smaller than 0.1. This needs to be checked with Matt.

⁴The long_name attributes are "LAMS Histogram Maximums" – this should be changed because one must find the peak from the 512-element array.

⁵Because the files produced by Matt did not have RAF-standard times, the "merge" chunk includes reference to a special version of getNetCDF called getNetCDFMH() that assigned the correct times so that the measurements could be merged with the Python-produced files.

Although the Python routine calculated horizontal wind variables and other results derived from the line-of-sight airspeeds, all such variables were recalculated in the present R routine (this file) to generate measurements of vertical and horizontal wind, airspeed, and angles of attack and sideslip. For airspeed, angle of attack, and sideslip, these steps were used:

1. The three-beam measurements were used to determine a three-dimensional relative wind vector $\mathbf{RW}=\{RW_x, RW_y, RW_z\}$ with components inbound along the longitudinal forward, lateral starboard, and lateral downward axis in the coordinate frame of the LAMS IRU (with attitude angles CPITCH, CROLL, CTHDG). The algorithm is documented in LAMSprocessing4Bwind.pdf. However, the three-beam version is used here.
2. The airspeed is then the magnitude of the vector \mathbf{RW} .
3. The angle of attack is $\alpha = \arctan(RW_z/RW_x)$ and the sideslip angle is $\beta = \arctan(RW_y/RW_x)$. These are in the reference frame of the LAMS (i.e., that defined by the CMIGITS IRU).

For measurements of wind, the R function `Ranadu::WindProcessor()` was used to calculate the wind based on LAMS-based variables for airspeed, angle of attack, and sideslip and the CMIGITS-provided attitude angles and ground-speed components. The complementary-filter adjustment provided by “`WindProcessor()`” is suppressed because the Kalman-filter processing in the CMIGITS accounts for the GPS-provided measurements of ground speed.⁶

The next step is then to run this routine, “`WindInSOCRATES.Rnw`”, in RStudio to generate the PDF document. That program contains a suppressed R chunk called ‘merge’ that should be executed on the first run to merge Matt’s variables into this file. After the first run, the program will load the merged data.frame from “`DataM.Rdata`”.

2.2 Results

The first comparison shown is the along-beam measurements of airspeed from LAMS in comparison to the wind vector from the standard wind-sensing system projected in the direction of the beams. For these plots, the relative-wind vector from the standard radome-based system is transformed to the Earth-relative l-frame and then to the reference frame of the LAMS, here called the g-frame by analogy to the gust-pod reference frame discussed elsewhere. This gives the relative-wind vector in the LAMS reference frame, which (like the a-frame) has x-axis forward, y-axis starboard and z-axis downward. The dot product of the unit vectors representing the orientations of the LAMS beams with the LAMS-relative wind then gives the expected along-beam airspeeds as predicted from the standard measurement of relative wind. Those unit vectors⁷ are:

$$\begin{pmatrix} \cos(35) & 0 & \sin(35) \\ 1 & 0 & 0 \\ \cos(35) - \sin(35)\sin(60) & -\sin(35)\cos(60) \\ \cos(35) & \sin(35)\sin(60) & -\sin(35)\cos(60) \end{pmatrix}$$

⁶Some adjustment may be needed to these values because I think the GPS antenna used is the standard one mounted below the fuselage and therefore separated from the LAMS pod.

⁷It would be useful to learn if there are better values for these angles for ARISTO-2017.

The results are shown in Figs. 1 and 2, after adjustments to the pitch and heading angles from the CMIGITS, each by $+2^\circ$. Without these shifts, there are obvious offsets in the cases of beams 1 and 3. This suggests that the LAMS IRU may not be oriented exactly along the LAMS axis. The plot for beam 4 is based on the approach that uses Savitzky-Golay polynomials⁸ because Matt did not produce beam-4 LOS values. These have not been used in the TAS calculation and were often missing.

The airspeeds obtained by the LAMS and by the standard pitot-tube-based wind sensing system are in excellent agreement, as shown in Fig. 3. A regression fit to TASX as a linear function of TASL gives an intercept coefficient of -0.12 m/s and a slope coefficient of 0.9994 , with a residual standard deviation of 0.43 m/s for 19,075 measurements from flights 1–4 of ARISTO-2017. These qualifications were imposed for a measurement to be accepted in these plots:

1. $TASX > 90$, to eliminate flight periods with possible flaps during takeoff and landing;
2. $lroll < 3^\circ$, to eliminate turns;
3. $0.001 < V_LOS_Beam2_Uncertainty < 0.1$, to eliminate the points found by Matt Hayman’s processing to have higher uncertainty, or in the case of value 0 to be missing.

```
lfTAS <- lm(TASX ~ TASL, data=DataR)
SummarizeFit(lfTAS)

## lm(formula = TASX ~ TASL, data = DataR)
## [1] "Coefficients:"
##           Estimate Std. Error  t value  Pr(>|t|)
## (Intercept) -0.12103 0.01801195   -6.7194 1.8759e-11
## TASL         0.99938 0.00010006  9987.4292 0.0000e+00
## [1] "Residual standard deviation: 0.428, dof=19073"
## [1] "R-squared 1.000"
```

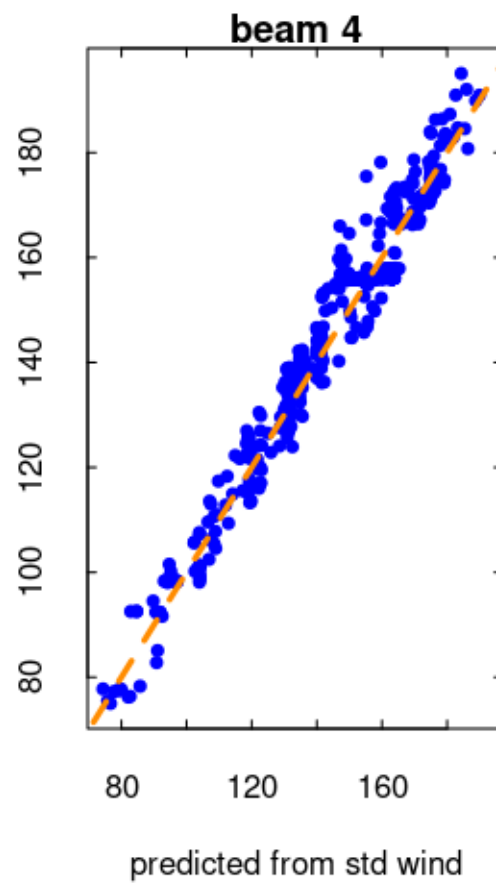
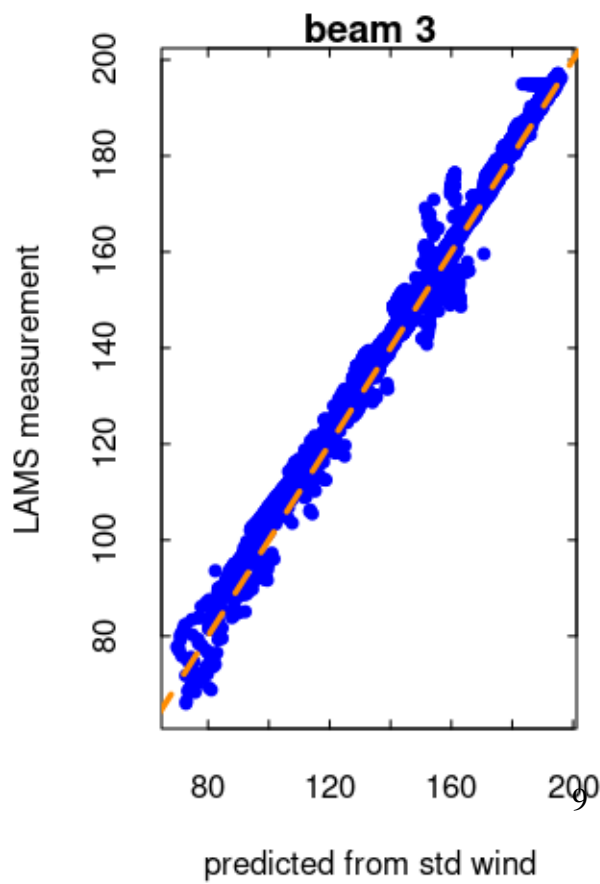
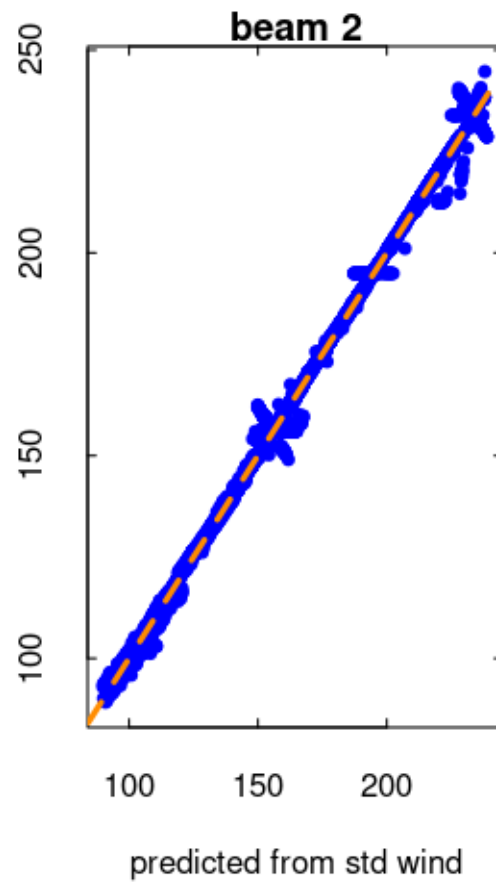
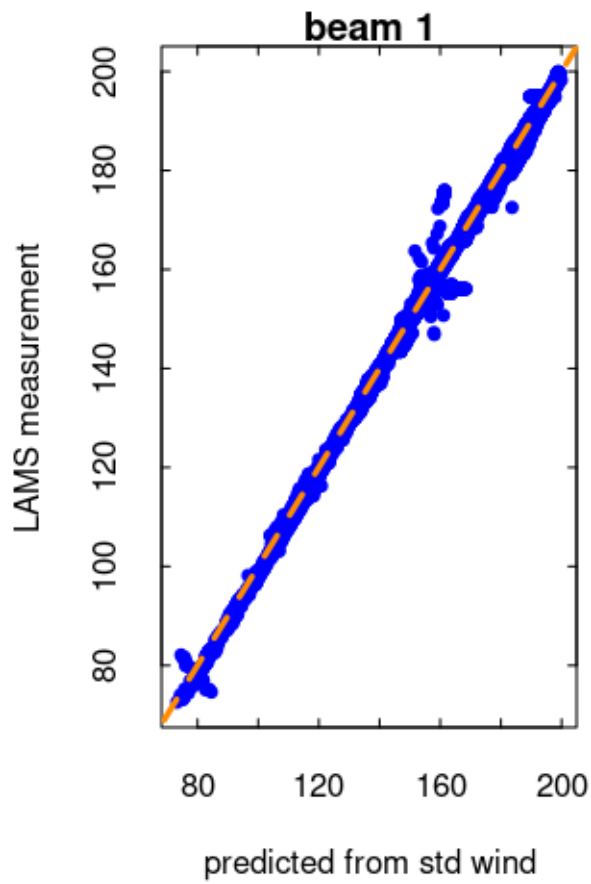
2.3 Recommendation: PCor function

Because the measurements TASX and TASL agree so well, it does not seem useful to adjust the “PCor” function for PSF and QCF. Instead, this check should be regarded as a test that the PCor correction remains appropriate, and it should be left at the standard formula for consistency with other projects.

3 Study of the Pitot-Static Sensor

New measurements were available during ARISTO-2017 from a pitot-static sensor designed to minimize the problems with high-frequency noise arising from line resonance in the standard

⁸Analysis using the S-G polynomials produced results in good agreement with Matt’s, but were more often missing, especially in cases where Matt assigns an uncertainty of more than 1 m/s.



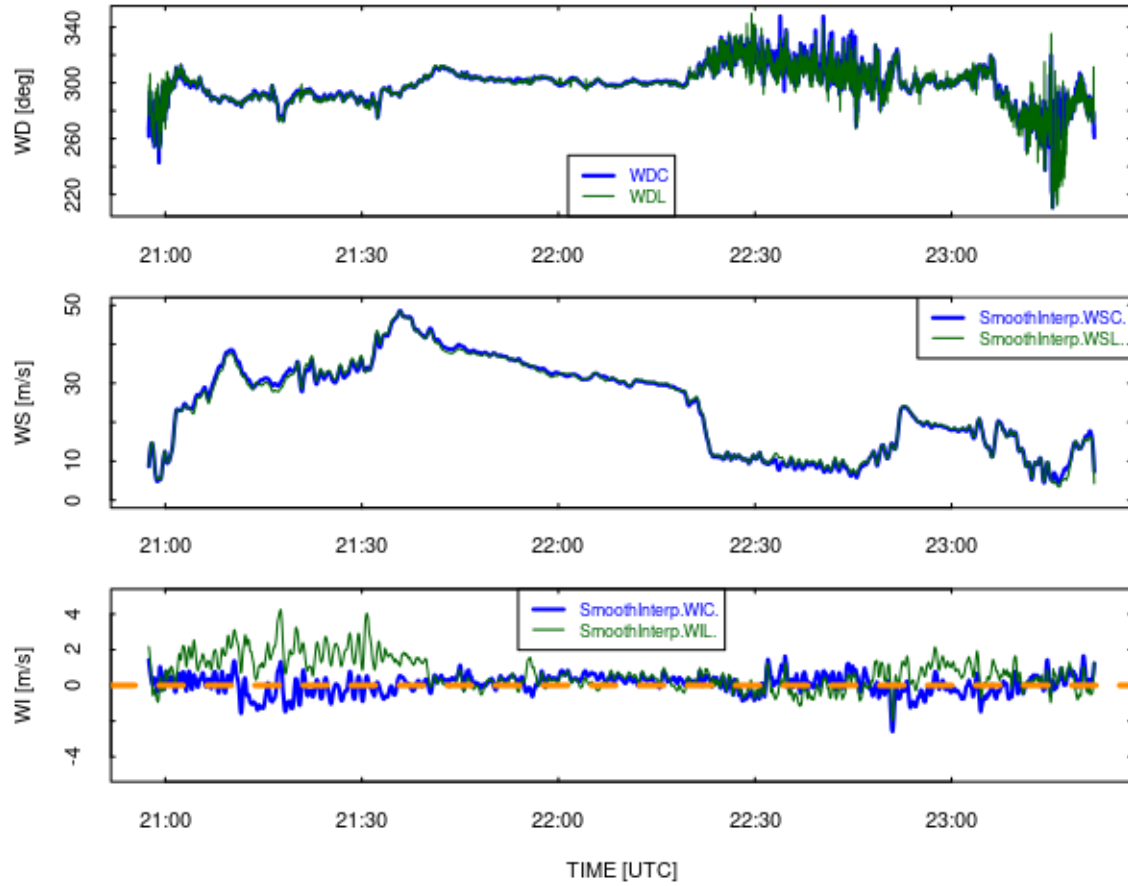


Figure 2: Wind measurements from ARISTO-2017 flight 3. The blue lines are from the standard wind-sensing system and the green lines are from LAMS.

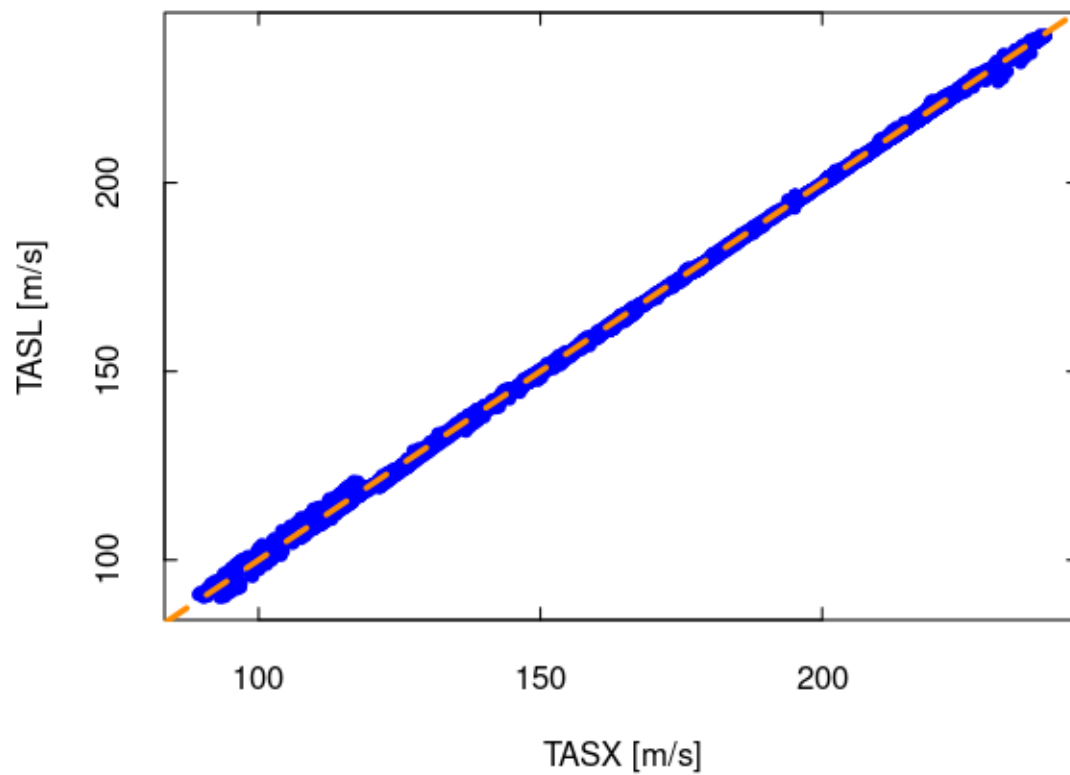


Figure 3: Comparison of measurements of airspeed by the conventional system (TASX) and by the LAMS (TASL). Restrictions have been applied to isolate a subset of measurements expected to be most reliable, as discussed in the text.

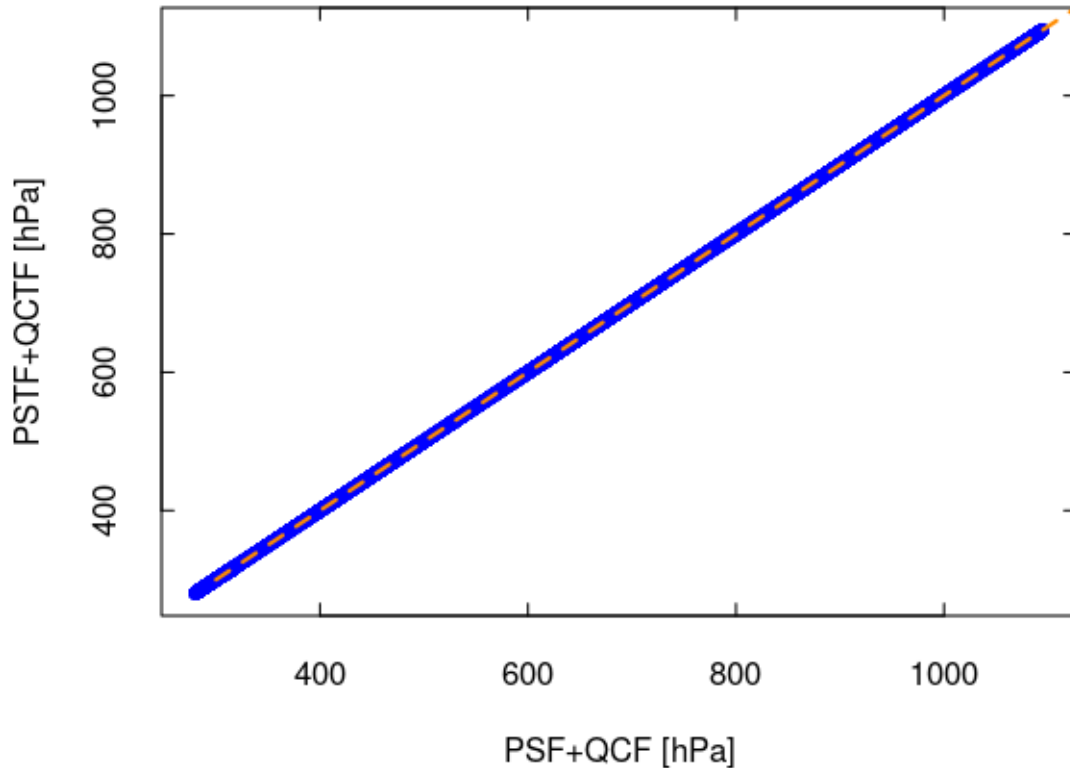


Figure 4: Total pressure measured by the pitot-static tube (PSTF+QCTF) vs the total pressure measured by the standard wind-sensing system (QCF+PSF). A Deming fit to these two variables gives an offset of 0.015 hPa and a slope coefficient of 1.000004, with a standard deviation between measurements of 0.14 hPa.

system. The new variables are QCTF and PSTF, representing the dynamic and static (ambient) pressure. Because the sensor is located in a region where flow distortion around the aircraft is severe, the static pressure differs significantly from the standard measurements including PSF and PSFC. However, the static pressure is not particularly useful because there is already a good measurement of static pressure available from the static-button ports and associated sensor. It is the measurement of dynamic pressure that is needed to improve high-frequency measurements of wind.

3.1 A proposed algorithm

Nevertheless, it is necessary to deal with the static-pressure measurement PSTF because QCTF is the difference between the measurements of total pressure at the stagnation point in the tube and the measured static pressure. The total pressure measurement is PSTF+QCTF, and it is in

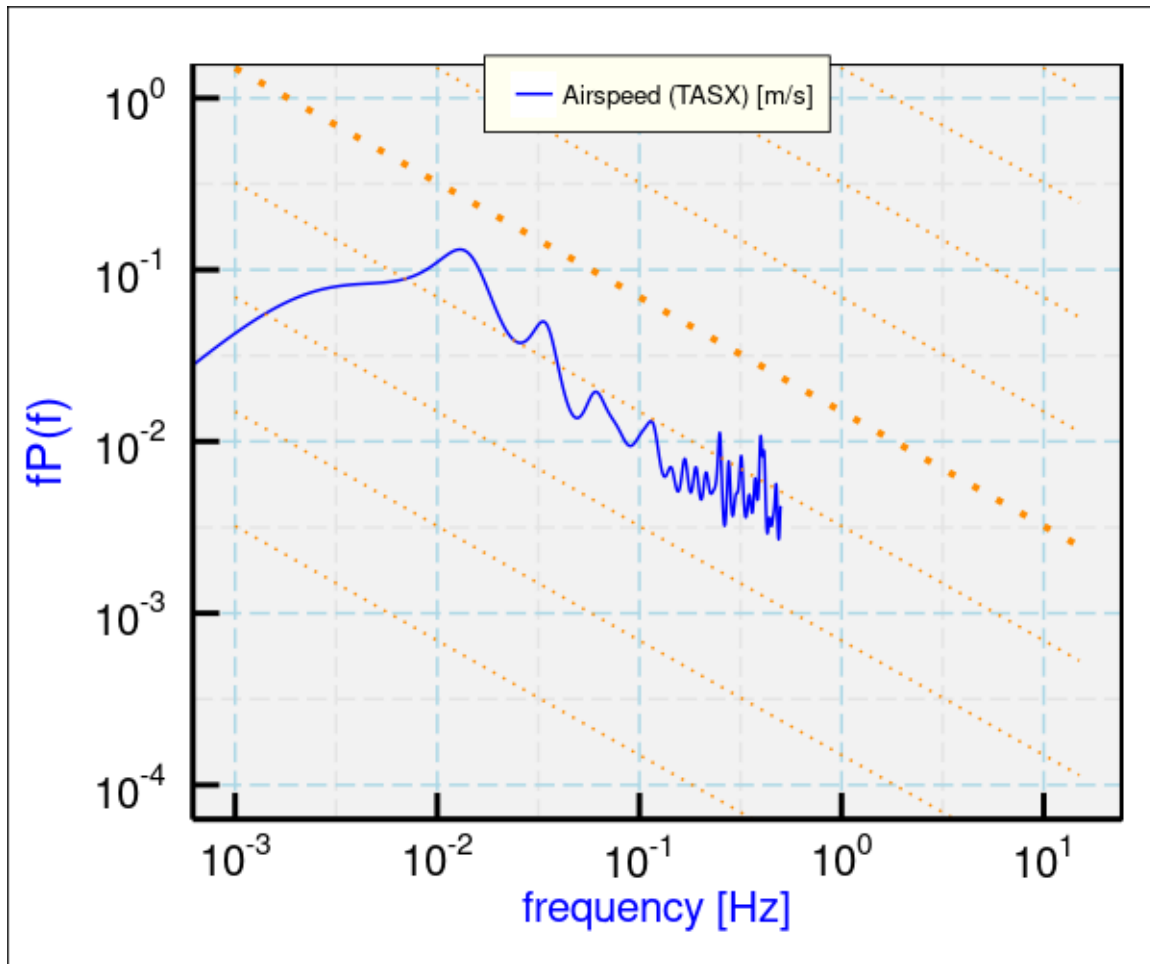


Figure 5: The variance spectrum for airspeed (TASX) for the period 4:00:00 – 5:00:00 UTC during SOCRATES flight 1.

remarkable agreement with PSF+QCF, as shown in Fig. 4. (The mean difference is 0.03 hPa; the sample standard deviation of that difference is 0.12.) While the pitot-static measurements could be approached like those of the gust-pod, seeking empirical equations that provide a match to QCFC, another approach is instead to subtract a smoothed version of the ambient pressure measured by the standard system (PSFC) from QCTF+PSTF to obtain QCTC.

It is useful to consider first the characteristics of the standard measurements of ambient, dynamic and total pressure. An example is a region with moderately low turbulence, 4:00:00–5:00:00 UTC on SOCRATES flight 1, which is characterized at high frequency by an eddy dissipation rate of about $10^{-5} \text{m}^2 \text{s}^{-3}$. The spectral variance for the airspeed measurement is shown in Fig. 5, and at low rate there are no evident problems. The corresponding plot for dynamic pressure (QCF or QCFC) looks similarly good. In contrast, the plot for the variable PSF has much lower amplitude, as shown in Fig. 6, and shows a clear noise floor corresponding to that expected for a random error of about 0.15 hPa. This matches the quoted uncertainty for the transducer (see “accuracy” at this URL). For measurements at frequencies above about 0.1 Hz, the measurement PSF is dominated by noise and does not provide useful information.

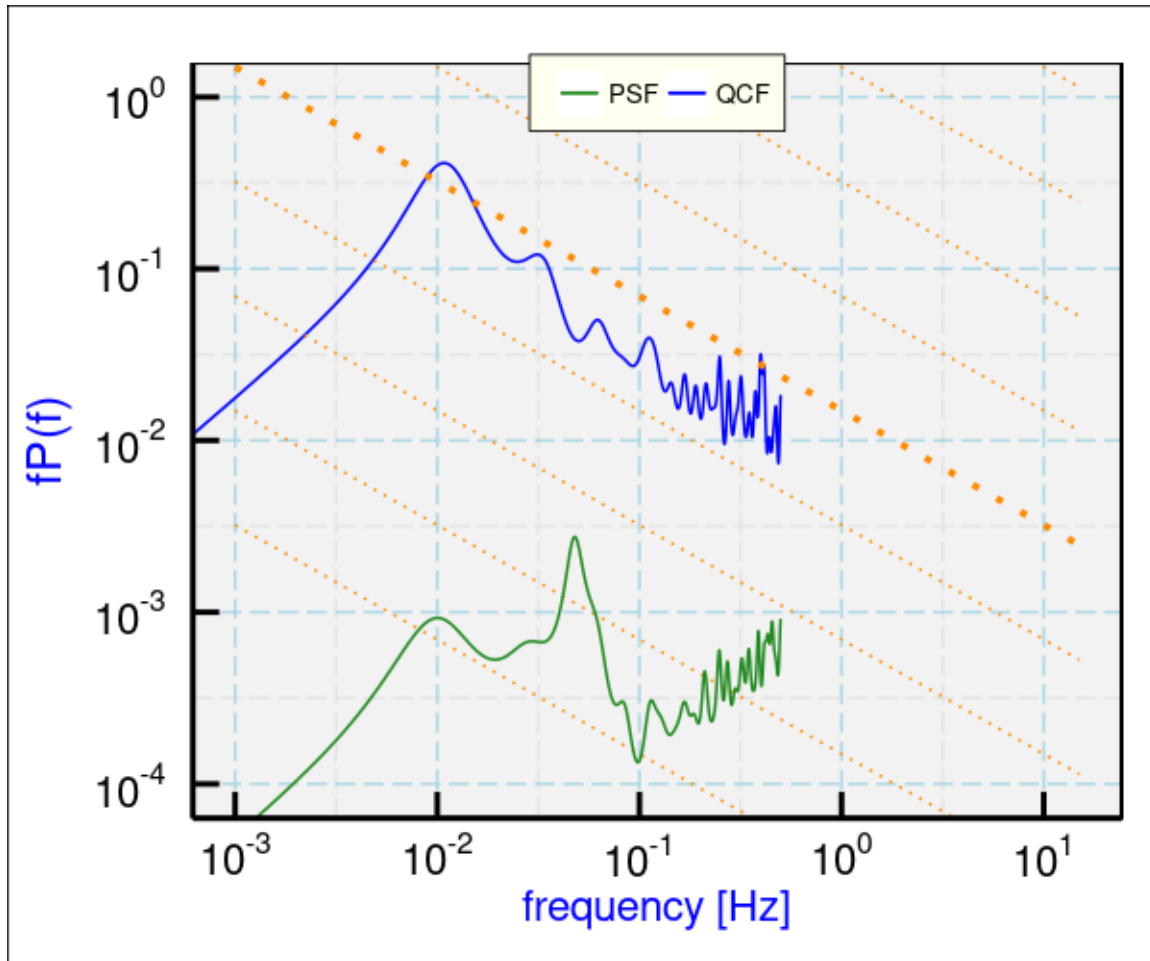


Figure 6: Variance spectra for QCF and PSF for the same period as in the previous plot.

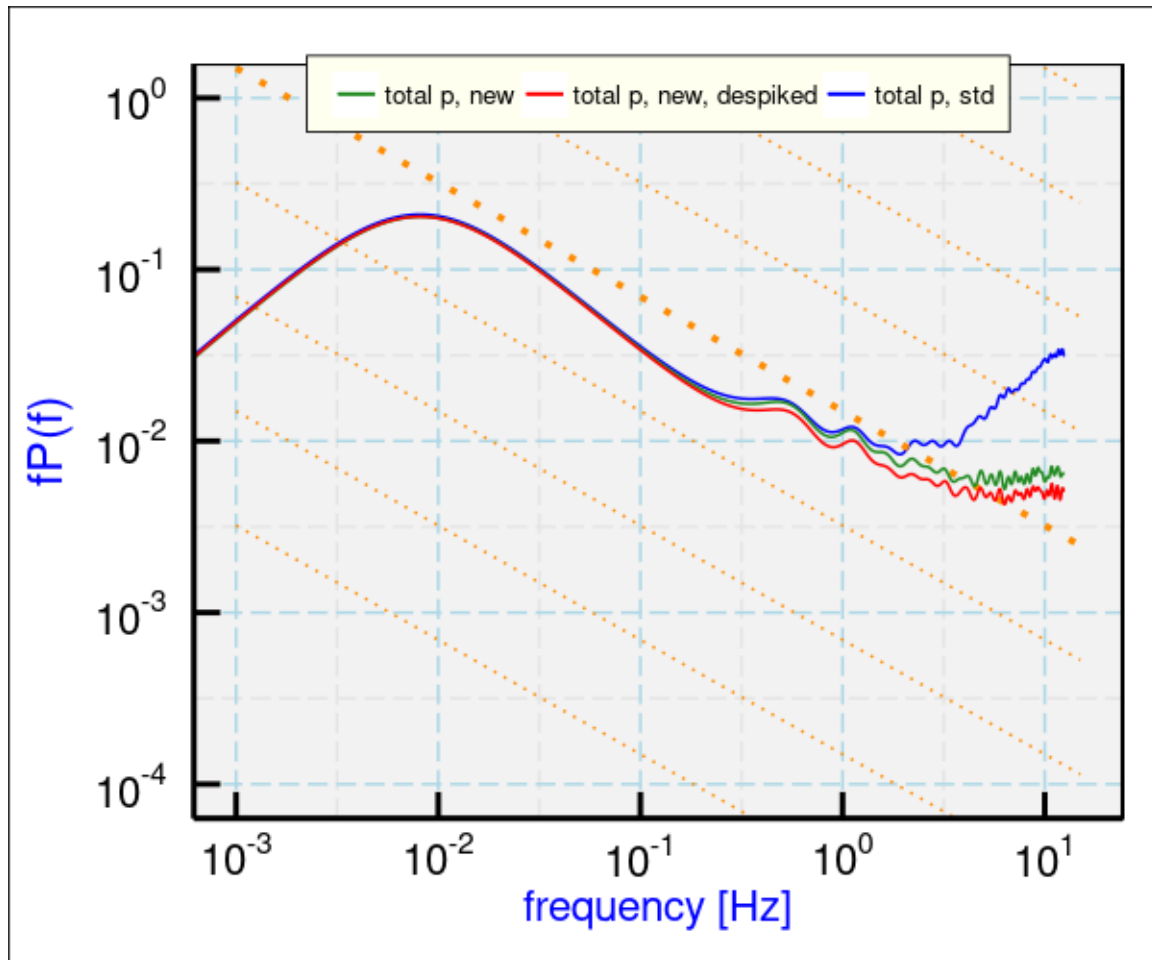


Figure 7: Measurements of total pressure (ambient plus dynamic) from the standard system and from the new pitot-static system. Data from SOCRATES flight #1, 4:00:00–5:00:00 UTC.

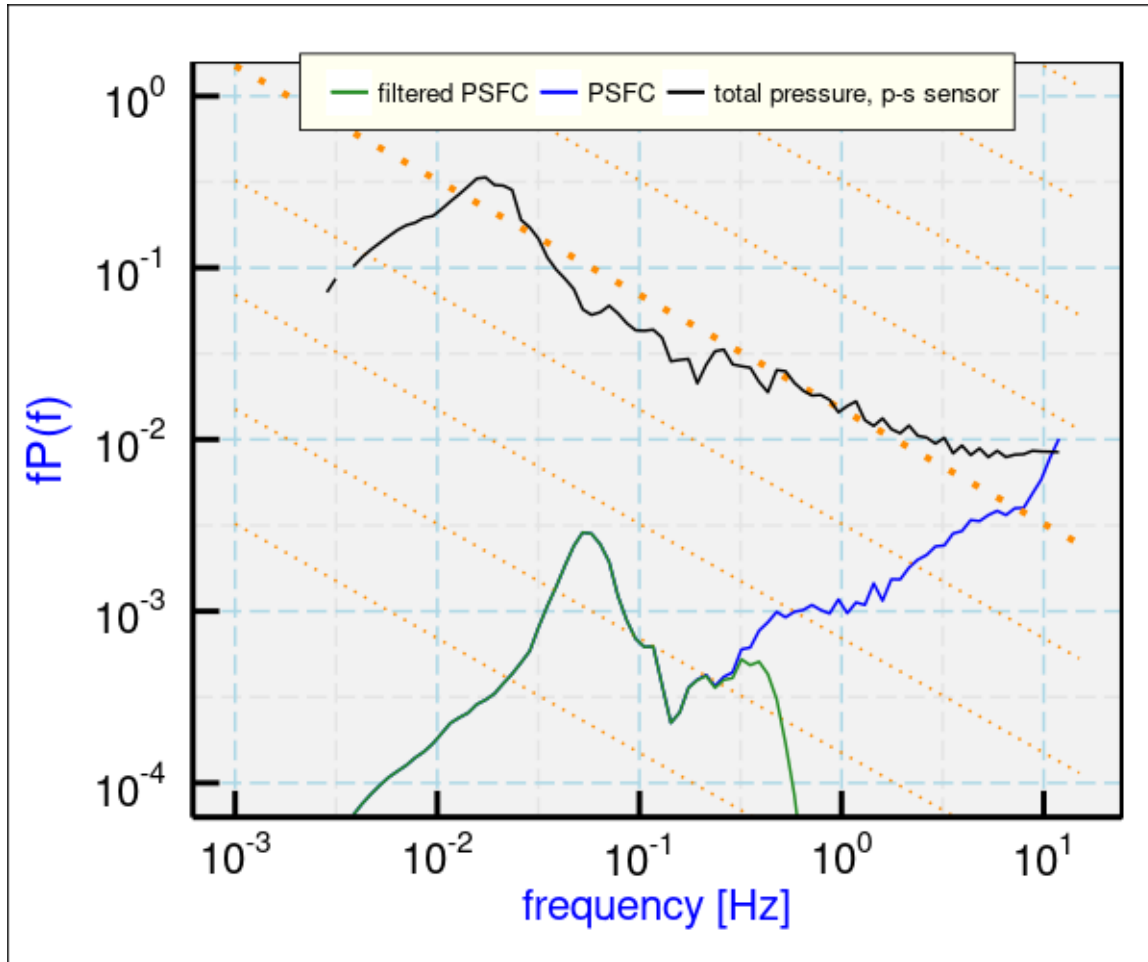


Figure 8: Filtered and unfiltered measurements of ambient pressure (PSFC) for SOCRATES Flight 1, 4:30:00 – 5:00:00 UTC. (Spectra are smoothed in 100 logarithmically spaced bins.)

Although the total pressure measurements from the standard and the pitot-static sensors agree well for 1-Hz measurements, as shown in Fig. 4, the two differ significantly in high-frequency response, as shown in Fig. 7, because there is much more noise in the standard measurements. The pitot-static sensor did not show a similar increase proportional to frequency, as expected for random noise, but it did show some apparent flattening in the frequency range above about 4_Hz. Both the standard and pitot-static measurements have some apparently spurious spikes in the time period used, but an aggressive elimination of possible spikes lowered the spectrum for the pitot-static sensor (red line in the plot) but did not remove the flattening. Similarly, elimination of spikes⁹ in the standard total-pressure measurement only lowered the noise spectrum slightly (by about 25%) but did not remove the strong evidence of noise in this measurement.

The point of this digression into the individual variance spectra is that, for the method proposed,

⁹The method used was to calculate the running mean and standard deviation of the variable, then remove measurements that differed more than two standard deviations from the mean and replace them with interpolated values. The usual test is five standard deviations, but using two is the “aggressive” test used here only for this purpose.

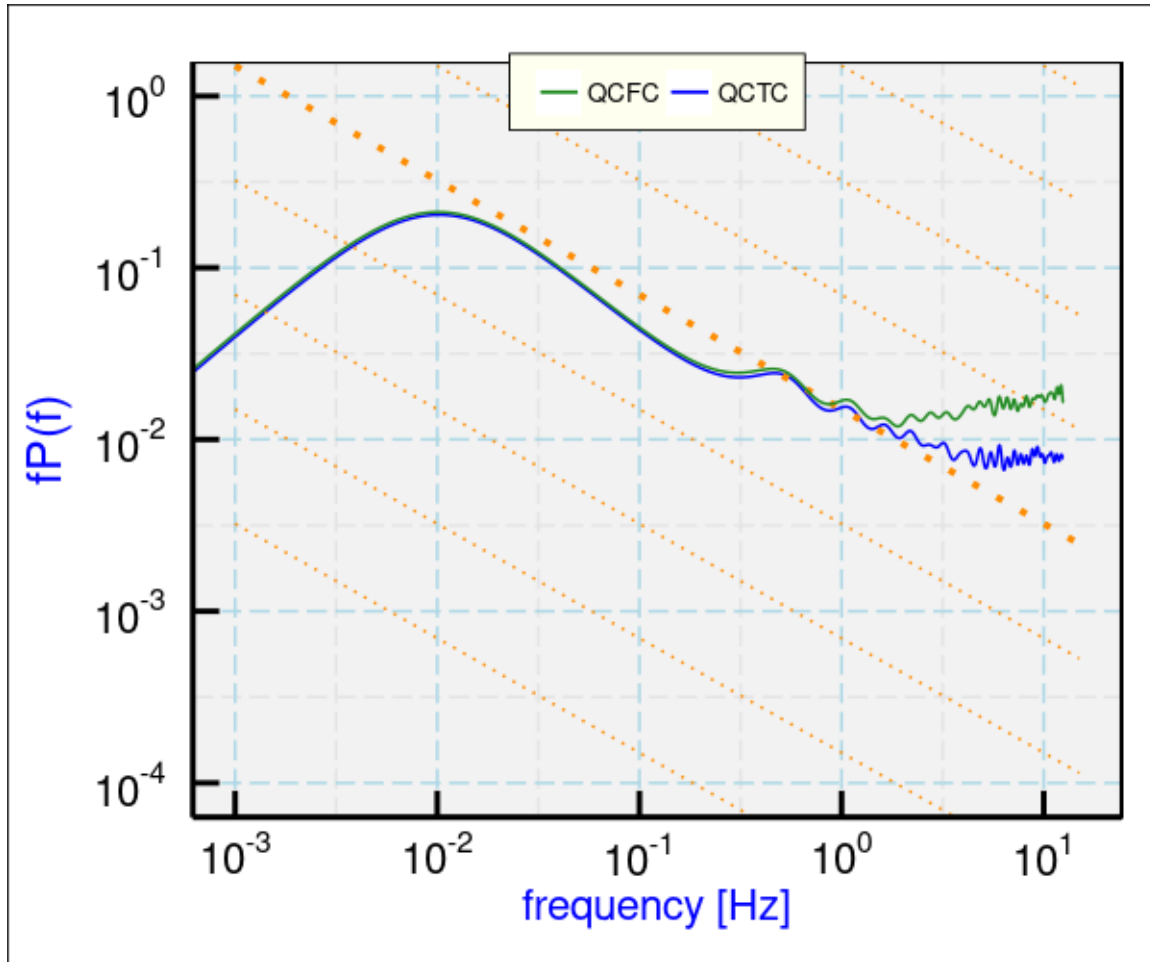


Figure 9: Variance spectrum for the variable QCTC, the corrected dynamic pressure from the pitot-static sensor. Also shown in the corresponding spectrum from the standard measurement (QCFC). The data are from 4:00:00 – 5:00:00 on SOCRATES Flight 1.

subtracting the standard pressure PSFC from the total pressure from the pitot-static tube will contaminate the result by inclusion of the noise in PSFC. Therefore, filtering to remove that noise is needed. A review of the 25-Hz data from SOCRATES Flight #1 indicated that PSF consistently exhibits a noise spectrum for frequencies above about 0.3 Hz, so a reasonably sharp low-pass filter with that frequency cutoff should eliminate little real signal while removing the contaminating noise. Figure 8 shows the result of filtering along with the unfiltered variable PSFC and the total pressure measurement from the pitot-static sensor (black line). The true signal present in PSFC is very small at the cutoff frequency (0.5 Hz), so nothing but noise is lost by using the filtered variable. Without such filtering, the variable PSFC would contaminate the derived dynamic pressure from the pitot-static sensor (here, QCTC) if it were to be calculated from $QCTC = QCTF + PSTF - PSFC$, although this could be used for low-rate measurements. At high rate (25 Hz), the following should be used: $QCTC = QCTF + PSTF - PSFF$ where PSFF is the result of a low-pass filter applied to PSFC.

The variance spectrum for the resulting dynamic pressure from the pitot-static tube is shown in

Fig. 9 as the variable QCTC. Also shown is the standard measurement from the radome-based system (QCFC) after all corrections (often called PCors) have been applied. No such correction is needed for QCTC because the appropriate corrections have already been applied to PSFC before it is used to calculate QCTC. The high-frequency portion of the new spectrum is much better than the standard one, but it still shows some probably spurious flattening beyond about 3 Hz. There are easily identified spikes in the high-rate time series being used, but the spike-removal algorithm did not remove this plateau. It is important to correct the cause of the spikes, though, because there may also be smaller ones not identified by the spike-removal algorithm that nevertheless may contaminate the high-rate variance spectrum.

Further discussion of the consequences for measurements of wind components will be deferred to a later section so that the gust-pod measurements can be discussed first and then all can be compared.

4 Using the Gust Pod in SOCRATES

Simple extension of the DEEPWAVE approach to the gust pod did not work well, so a different approach is developed in this note. The key problem arises from the gust-pod measurement of the relative wind, because the quantity needed is the relative wind with respect to the aircraft, not the gust pod. The gust pod is located in a region of distorted airflow where the sideslip and angle-of-attack components may affect each other, the pod is not oriented along the aircraft axis and may move relative to the fuselage. The wind is measured by combining the measurement of relative wind vector, transformed to a reference frame fixed relative to the Earth, with the measurement of the aircraft velocity vector. With the inclusion of the large pod in SOCRATES, the airflow distortion has become more problematic.

4.1 The general approach

The CMIGITS INS used by the gust pod performs well and provides appropriate measurements of the gust-pod motion relative to the Earth to use when calculating the wind. Small corrections required when the pitch or roll angles change, leading to motion of the gust pod relative to the aircraft reference frame, initially will be ignored. The differences in aircraft-velocity components between the two systems are typically quite small: For SOCRATESf01, the mean difference between velocity components was 0.0005 ± 0.05 m/s (sample standard deviation) for the easterly component and -0.0002 ± 0.06 m/s for the northerly component. Most of the listed sample standard deviation was contributed by periods when the roll angle was outside $\pm 3^\circ$; within those limits, the sample standard deviation for 1-Hz measurements was about 0.025 m/s. Therefore, for the purpose of the present analysis, no transformation or correction to the CMIGITS-provided measurements of aircraft velocity is needed except for the omitted compensation for the motion of the gust pod relative to the Honeywell INS. The gust pod is located 13 m behind the INS, 7 m to the port side, and within about 1 m vertically. However, the GPS antenna that the CMIGITS INS uses is at the same location as that used by standard

wind processing, so the CMIGITS aircraft-velocity components are updated to that reference point. Mean updated measurements of ground-speed components should therefore agree.

These steps then are involved in the new processing scheme:

1. Define a new reference frame, here called the g-frame. Three reference frames then are used here: the a-frame (defined with \hat{x} forward along the longitudinal axis of the aircraft, \hat{y} to starboard and perpendicular to \hat{x} , and \hat{z} determined by the cross product of \hat{x} and \hat{y} but generally downward); the g-frame (defined analogously but with reference to the gust-pod orientation which is slightly different from the aircraft reference frame); and the l-frame (the local Earth-relative frame with \hat{x} eastward, \hat{y} northward, and \hat{z} upward). Transformation among these reference frames was described in the Workflow Document for the Kalman Filter Technical Note. Because the CMIGITS IRU aligns with some residual error that is gradually corrected in flight by Kalman-filter updating, and because there is some motion of the wings in turbulence and as the aircraft fuel weight decreases, the difference in orientation between the two systems (Honeywell and CMIGITS) may change during the flight, so it is necessary to consider the attitude-angle transformation each time interval during processing rather than via a constant correction for the entire flight.
2. Estimate the relative-wind components in the a-frame using gust-pod measurements in the g-frame. In the g-frame, the relative wind that would be measured is distorted unacceptably from the true relative wind outside the flow-distortion field of the aircraft, so the solution is not to simply transform the measured relative-wind vector from the g-frame to the l-frame. Instead, the approach here is as follows:
 - (a) Consider the two differential-pressure measurements from the up-down and left-right pairs of ports on the gust pod as a two-component vector measurement representing the pressure difference at the gust pod. Make the trial assumption that the ports align with the axes defined by the CMIGITS INS. Transform this pressure-gradient vector from the g-frame to the a-frame, where it is resolved into \hat{x}_a and \hat{y}_a components.
 - (b) Find the best representation of the a-frame relative wind (approximately $\{V, V\beta, V\alpha\}$ with V the airspeed, β the sideslip angle and α the angle of attack) [XXX correct the signs] by a fit to the normal a-frame relative wind as a function of standard measurements and the transformed pressure-gradient components. This will give a relative-wind vector analogous to the one from the standard wind-sensing system but potentially independent of the radome pressure measurements that are sometimes affected by line freezing or contamination in the lines. This preserves the backup potential of the gust pod.
 - (c) It is also desirable to produce an airspeed measurement from the gust pod (TASG) that is independent of the standard system, but it is probably best to use TASX to calculate the relative wind because it is expected to be the best available measurement of airspeed in the a-frame. The “PCor” function used to adjust the airspeed measurement for the static defect depends on a valid measurement of angle of attack, but

in a relatively insensitive way, and normal processing substitutes an average value when no valid measurement of AKRD is available. TASG might provide a valuable backup in that case.

When the aircraft rolls about its longitudinal axis, complicated changes occur to the CMIGITS attitude angles. A pure roll change in the a-frame can change all three attitude angles in the g-frame because the gust-pod longitudinal axis is not aligned along the aircraft longitudinal axis. The angle transformations must account for this or the relative-wind measurements will not be valid for general orientations of the aircraft.

4.2 Details of the Transformation

The vector to be transformed has components (0, BDIF_GP, ADIF_GP) in the g-frame. Because in the R package “Ranadu” [<https://github.com/WilliamCooper/Ranadu.git>] there is a transformation function (XformLA) that transforms from the a-frame to the l-frame or the l-frame to the a-frame, that will be used here because that transformation function was tested extensively during the development of the Kalman filter. Details are described in the workflow document for the Kalman filter Technical Note (Cooper [2017]). The procedure is to transform the pressure vector from the g-frame to the l-frame using the CMIGITS-provided attitude angles, then transform from the l-frame to the a-frame using the Honeywell-provided attitude angles and the inverse function in XformLA(). The code listed below produces, in diffPa, the matrix of measured pressure differences transformed to the a-frame:

```
diffP <- c(rep(0, nrow(Data)), Data$BDIF_GP, Data$ADIF_GP) ## check signs!!
dim(diffP) <- c(nrow(Data), 3)
DF <- data.frame(ROLL=Data$CROLL_GP, PITCH=Data$CPITCH_GP, THDG=Data$CTHDG_GP)
diffPl <- Ranadu::XformLA (DF, diffP) ## DF contains gustpod attitude angles
diffPa <- Ranadu::XformLA (Data, diffPl, .inverse=TRUE) # pressure vector in a-frame
```

4.3 Empirical Coefficients

Angle of Attack

The standard method of calibrating the measurement of angle-of-attack is described in detail by Cooper et al. [2016], and a revised approach is described in Sect. 1.2. The latter is used here. The goal then is to find an empirical equation that duplicates that angle of attack (here, AK) but uses measurements from the gust-pod. Because of flow distortion at the gust-pod, it is expected that the empirical relationship will be more complicated and will involve the horizontal as well as vertical pressure differences from the gust-pod. In addition, the data used for the fit should include various maneuvers in addition to the speed runs normally used for finding a representation of AKRD.

The data set used here is the combination of SOCRATES test flight #1, ferry flight #2, and research flight #1. Test flight 1 included a complete set of maneuvers, so that flight provides

a good range of conditions for the fits. Research flight #1 included a low-level flight segment that is also important to include for characterization of this part of the flight envelope. The new variable will be called AK_GP and will be determined by linear fits to variables from the gust-pod after transformation to the a-frame.

In addition, an approach analogous to that for the complementary-filter representation of AKRD will be used because it provided an improved representation of the angles of attack and sideslip. The same approach is taken: Split the reference AKRD into “slow” and “fast” components, represent them separately using empirical functions of variables that are similarly filtered, and then combine them to produce the result for AK_GP. The following table shows the residual standard deviation for some choices of the empirical equation. The variables used are: Δp_z : the pressure difference between top-and-bottom pressure ports on the gust-pod after transformation to the a-frame; Δp_y : the corresponding difference between the starboard and port-side pressure ports; q_c : the dynamic pressure measured at the forward port on the gust-pod; and M_g : the Mach number calculated using the uncorrected measurements QC_GP and PS_GP from the gust pod.

Equation	residual standard deviation [°]
$a_0 + a_1 \frac{\Delta p_z}{q_g}$	0.166
” + $a_2 M_g \frac{\Delta p_z}{q_g}$	0.145
” + $a_3 \frac{\Delta p_y}{q_g}$	0.085
” + $a_4 \left(\frac{\Delta p_z}{q_g} \right)^2$	0.078

The resulting fit has residual standard deviation similar to that obtained in typical calibrations leading to AKRD, and as shown in Fig. 10 the five-coefficient fit provides a very good representation of AKRD. Figure. 11 shows an expanded view of the period of the speed run. The difference at large angle-of-attack (near 5°) is a little larger than elsewhere, but values of the angle-of-attack this large are seldom encountered except during maneuvers like this.

Sideslip

Sideslip was handled in a similar manner, except that the reference SSRD was the standard variable calculated without separating into slow and fast components because there does not appear to be a varying offset to the standard sideslip angle as there is for angle-of-attack. That is not true for the gust-pod sideslip, however, so a complementary-filter representation was used for it as for angle-of-attack. Some exploration of possible variables to include in an empirical representation led to this equation:

$$\beta = e_0 + \frac{\Delta p_y}{q_g} (e_0 + e_1 M_g) + e_2 \frac{\Delta p_z}{q_g} \quad (3)$$

which was used to fit the slow and fast components of SSRD, except for the fast component the constant e_0 was omitted. The residual standard deviation was 0.06° for the set of three flights (tf01, ff02, rf01). The good correspondence between the radome-based sideslip angle and the

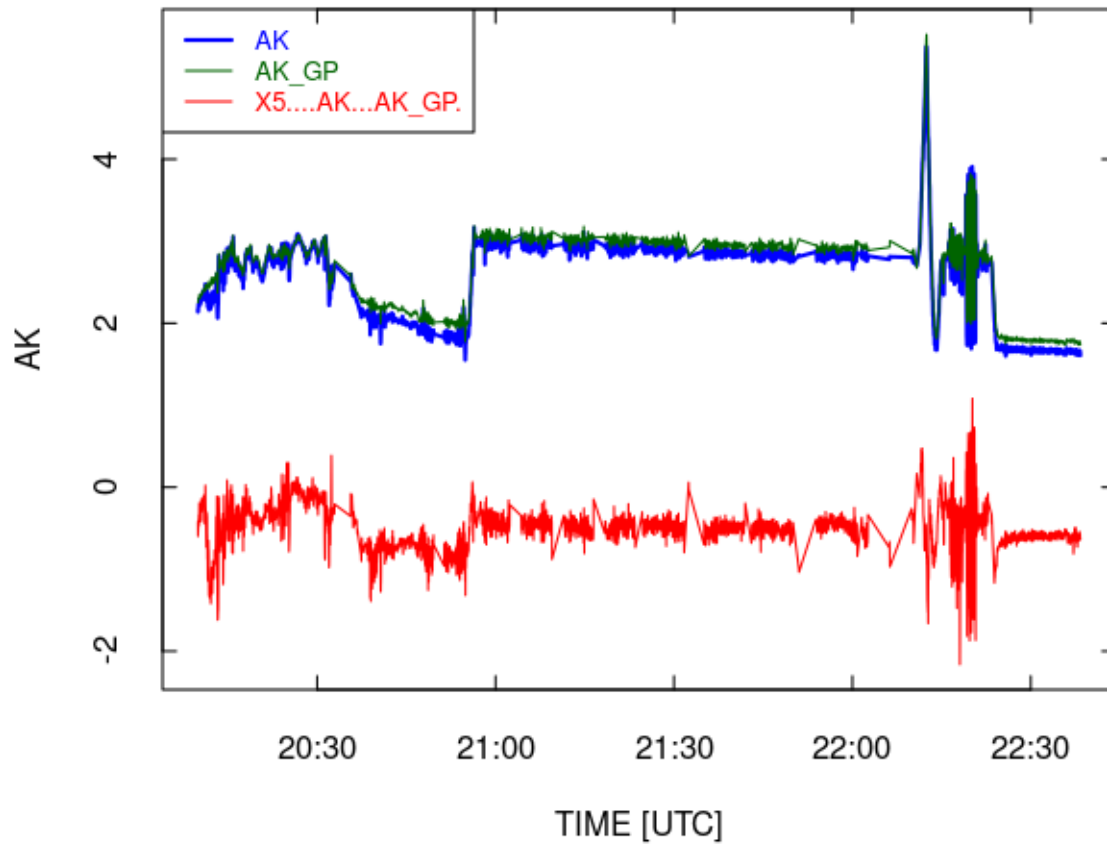


Figure 10: Comparison of gust-pod fit result (AK_GP) to the complementary-filter radome-based value for angle of attack (AK), for SOCRATES test flight 1. Also shown as the red line is the difference multiplied by 5 to make the difference easier to interpret.

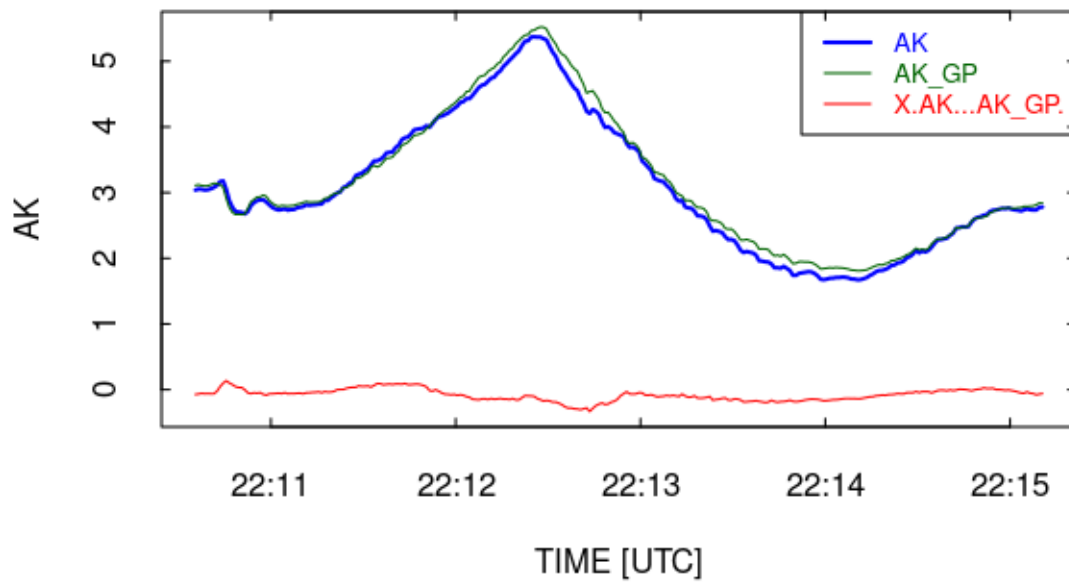


Figure 11: As in the preceding plot but expanded to show only the period of the speed run and without factor-of-five magnification of the difference (red line).

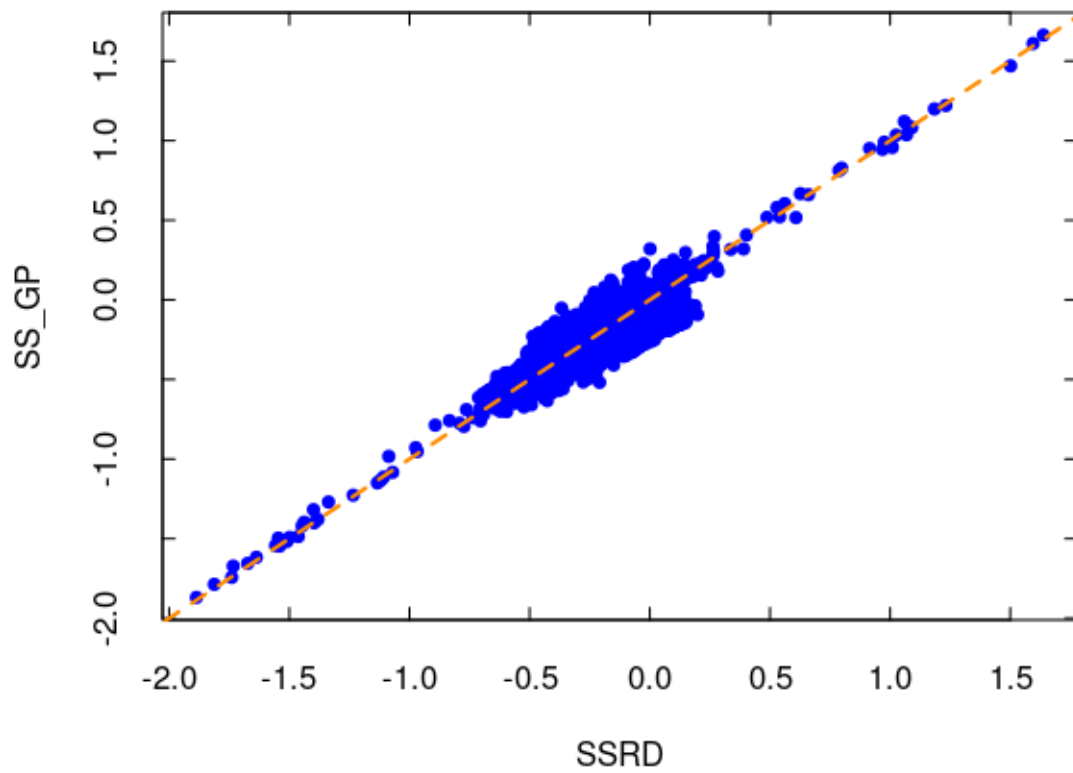


Figure 12: Comparison of SS_GP, calculated using the new SOCRATES coefficients, to SSRD from the radome, for flights tf01, ff02, and rf01 (with some short omissions of obviously problematic measurements).

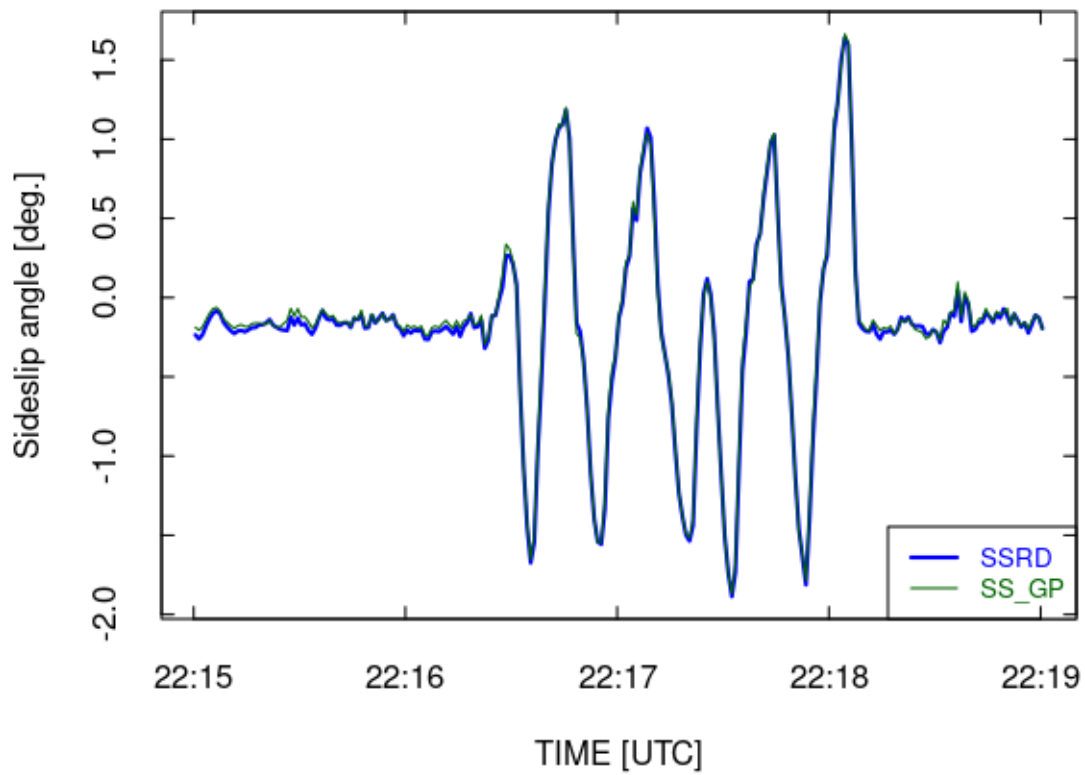


Figure 13: Comparison of SS_GP, calculated using the new SOCRATES coefficients, to SSRD from the radome, for the yaw maneuver from SOCRATES tf01.

gust-pod-based sideslip angle is shown in Fig. 12, and the yaw maneuver from SOCRATES tf01 (22:16:00–22:18:30) is shown in Fig. 13. The measurements from the radome and gust pod are in remarkable agreement through the period of this yaw maneuver.

Airspeed

The equation developed in Cooper et al. (2016) for airspeed was based on this fit to the ratio q/p as measured by the standard system:

$$\frac{q}{p} = d_0 + d_1 \frac{q_g}{p_g} + d_2 M_g \frac{q_g}{p_g} + d_3 \left(\frac{q_g}{p_g} \right)^2 + d_4 \frac{\Delta p_{\alpha,g}}{q_g} + d_5 M_g + d_6 \left(\frac{\Delta p_{\alpha,g}}{q_g} \right)^2 + d_7 M_g \frac{\Delta p_{\alpha,g}}{q_g} \quad (4)$$

where q is dynamic pressure, p is ambient pressure, M_g is the Mach number determined from the gust-pod measurements of dynamic and ambient pressure (q_g and p_g), and $\Delta p_{\alpha,g}$ is the pressure difference between top and bottom pressure ports on the gust-pod. The DEEPWAVE fit found coefficients $\{d_i\} = \{1.1940, 27.2893, -29.0483, 13.6231, -0.5936, -7.5812, 0.1554, 1.2607\}$. Repeating the fit for the same data set used to determine coefficients to represent the angles of attack and sideslip gave new coefficients as listed below and resulting predictions for q/p as shown in Fig. 14. A reasonable fit was obtained, but the high-frequency spectral variance of the resulting airspeed had apparent weaknesses so an approach similar to that used for angle-of-attack was used for q/p : Split all the linear-predictor terms (like q_g/p_g) into complementary low-pass and high-pass components, fit separately, and combine the results. While the fit is very good, it is significantly different from that found in the DEEPWAVE project. The fit coefficients for the low-pass and high-pass terms are listed below (where the numbered terms in the fit correspond respectively to the numbered coefficients in (4)).

```
## [1] "Fit to the low-pass-filtered components:"
## lm(formula = QRS ~ DVT2S + DVT3S + DVT4S + DVT5S + DVT6S + DVT7S +
##     DVT8S, data = DataF)
## [1] "Coefficients:"
##           Estimate Std. Error  t value  Pr(>|t|)
## (Intercept) -0.0085467  0.0013545  -6.3101 2.8148e-10
## DVT2S       -6.5399427  0.0750555  -87.1347 0.0000e+00
## DVT3S       13.5749628  0.1203449  112.8005 0.0000e+00
## DVT4S       -8.2527369  0.0781331 -105.6240 0.0000e+00
## DVT5S       -0.4159055  0.0033729 -123.3094 0.0000e+00
## DVT6S        1.0019495  0.0132433   75.6572 0.0000e+00
## DVT7S        0.1627284  0.0023608   68.9308 0.0000e+00
## DVT8S        1.0382988  0.0069614  149.1502 0.0000e+00
## [1] "Residual standard deviation: 0.004, dof=45241"
## [1] "R-squared 0.999"
## [1] "-----"
## [1] "Fit to the high-pass-filtered components:"
## lm(formula = QRF ~ DVT2F + DVT3F + DVT4F + DVT5F + DVT6F + DVT7F +
##     DVT8F - 1, data = DataF)
## [1] "Coefficients:"
##           Estimate Std. Error  t value  Pr(>|t|)
## DVT2F  -1.430651  0.1196579  -11.956  6.7638e-33
## DVT3F   4.228605  0.2011324   21.024  1.1734e-97
```

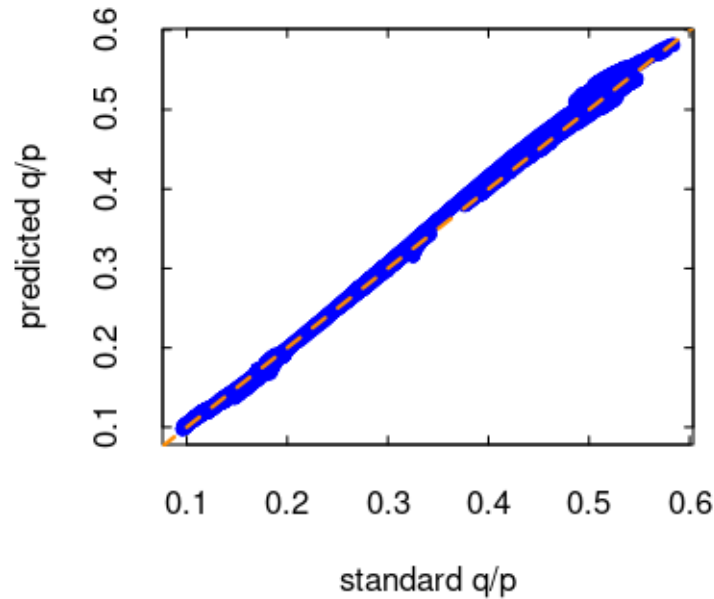


Figure 14: Predicted ratio q/p from gust-pod measurements vs that measured by the standard wind-sensing system, for the composite data from tf01, ff02 and rf01.

```
## DVT4F -1.866424  0.1288761  -14.482  2.0059e-47
## DVT5F -0.180203  0.0013908  -129.564  0.0000e+00
## DVT6F  0.375896  0.0159225   23.608  1.7937e-122
## DVT7F  0.052324  0.0015334   34.124  5.6225e-252
## DVT8F  0.487096  0.0031993  152.252  0.0000e+00
## [1] "Residual standard deviation: 0.000, dof=44847"
## [1] "R-squared 0.996"
```

Examples of Results

This processing recommended in the final section of this report was applied to SOCRATES research flight #1 so that the gust-pod measurements of wind can be compared to the standard wind measurements based on the radome. Figure 15 shows that the horizontal wind from the gust pod matches that from the standard system quite well.¹⁰ The wind measurements from the gust pod, entirely independent of variables provided by the radome, provide a valid measurement of the wind that is in reasonable agreement with the standard system and can provide a back-up measurement in cases where some problem interferes with the measurements from the standard system.

¹⁰The large spikes are associated with periods of missing gust-pod data. The spike-removal algorithm discussed in Sect. 5.2 has not been used when constructing these plots.

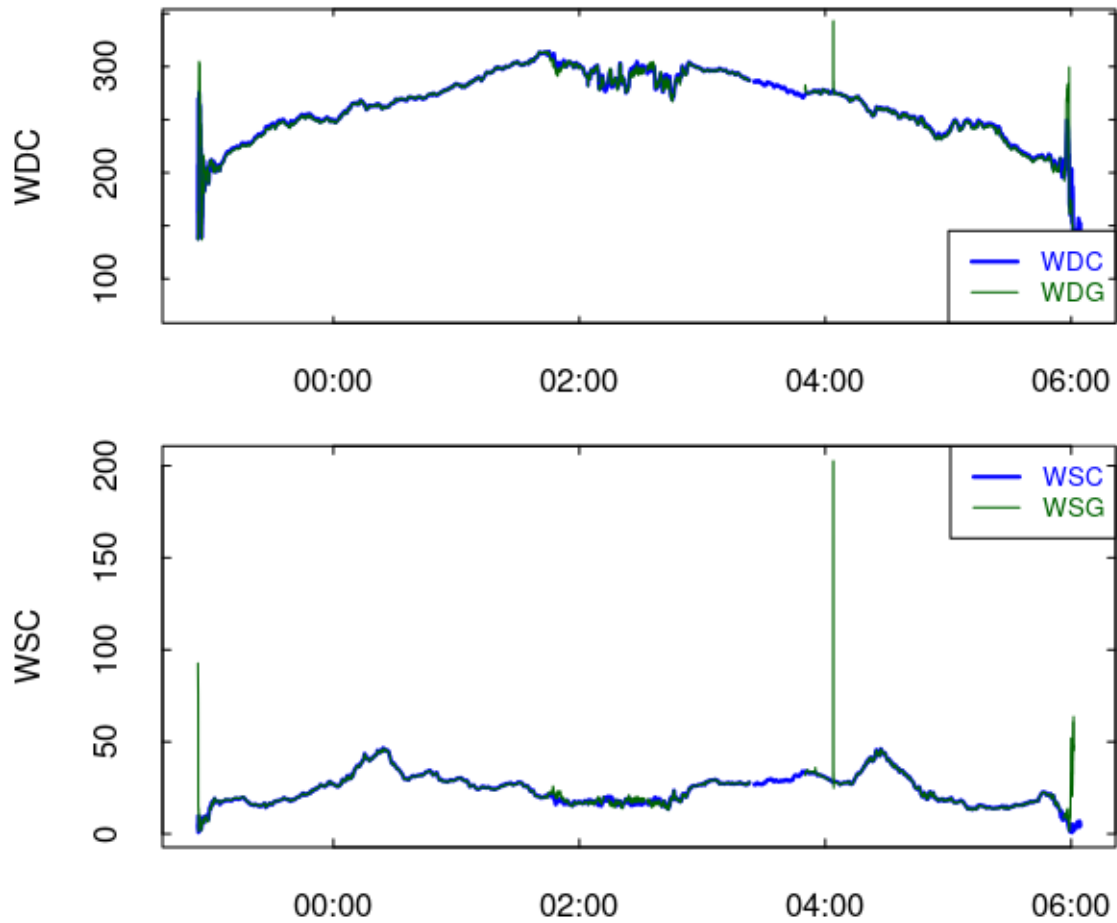


Figure 15: Comparison of measurements of the horizontal wind from the standard calculation (WDC and WSC) and from the gust pod (WDG and WSG).

4.4 Summary

Use these equations and coefficients to find the relative-flow angles and ratio of dynamic to ambient pressure (with MACHGG, different from MACHG defined below, defined as the Mach number calculated from QC_GP and PS_GP). The variables DIFFPY and DIFFPZ are defined earlier as the pressure-difference components measured by the gust pod and then transformed to the a-frame. These definitions are also used: ARG=ADIF_GP/QC_GP and QRG=QC_GP/PS_GP). Because the coordinate-frame transformations and extensive splitting for complementary representation of the results are complex, it seems preferable to add these variables in second-pass processing. That is also a convenient way to add the complementary-filter result for AKRD and WIC (here, AK and WIY).

In the following summary, the function “bf()” denotes the low-pass Butterworth filter and “cf()” is the complementary high-pass filter.

$$\begin{aligned} \mathbf{AK_GP}: & 2.6143 + 7.7777 * \text{bf}(\text{DIFFPZ} / \text{QC_GP}) + \\ & -7.3179 * \text{bf}(\text{DIFFPZ}/\text{QC_GP} * \text{MACHGG}) + \\ & 2.5172 * \text{bf}(\text{DIFFPY}/\text{QC_GP}) + \\ & -3.8262 * \text{bf}(\text{DIFFPZ} / \text{QC_GP})^2 + \\ & 5.7670 * \text{cf}(\text{DIFFPZ}/\text{QC_GP}) + \\ & -1.9574 * \text{cf}(\text{DIFFPZ}/\text{QC_GP} * \text{MACHGG}) + \\ & 0.8618 * \text{cf}(\text{DIFFPY}/\text{QC_GP}) \end{aligned}$$

$$\begin{aligned} \mathbf{SS_GP}: & -1.39231 + 11.17897 * \text{bf}(\text{DIFFPY}/\text{QC_GP}) + \\ & -10.68744 * \text{bf}(\text{DIFFPY}/\text{QC_GP} * \text{MACHGG}) + \\ & -2.15706 * \text{bf}(\text{DIFFPZ}/\text{QC_GP}) + \\ & 15.7052 * \text{cf}(\text{DIFFPY}/\text{QC_GP}) + \\ & -6.3512 * \text{cf}(\text{DIFFPY}/\text{QC_GP} * \text{MACHGG}) + \\ & -3.0271 * \text{cf}(\text{DIFFPZ}/\text{QC_GP}) \end{aligned}$$

$$\begin{aligned} \mathbf{QP_GP}: & -0.0085467 + \\ & \text{bf}(\text{QRG}) * (-6.53994 + 13.57496 * \text{bf}(\text{MACHGG}) + \\ & -8.25274 * \text{bf}(\text{QRG})) + \\ & \text{bf}(\text{ARG}) * (-0.41591 + 0.16273 * \text{bf}(\text{ARG}) + 1.0383 * \text{bf}(\text{MACHGG})) + \\ & 1.00195 * \text{bf}(\text{MACHGG}) + \\ & \text{cf}(\text{QRG}) * (-1.43065 + 4.22861 * \text{cf}(\text{MACHGG}) + \\ & -1.86642 * \text{cf}(\text{QRG})) + \\ & \text{cf}(\text{ARG}) * (-0.1802 + 0.05232 * \text{cf}(\text{ARG}) + 0.4871 * \text{cf}(\text{MACHGG})) + \\ & 0.3759 * \text{cf}(\text{MACHGG}) \end{aligned}$$

MACHG: Calculate from QP_GP:

$$\text{MACHG} = \sqrt{\frac{c_v}{R_a} ((1 + \text{QP_GP})^{R_a/c_p} - 1)}$$

where c_p , c_v , R_a are respectively the specific heat at constant pressure, specific heat at constant volume, and gas constant for air. The appropriate respective values for dry air are {1004.728, 717.6631, 287.07}.

- TASG:** Calculate from MACHG and ATX (optionally with humidity correction, here and in the calculation of MACHG).
- Wind:** Calculate using gust() but with TASG, CVEW_GP, CVNS_GP, CVSPD_GP, CPITCH_GP, CROLL_GP, CTHDG_GP, AK_GP, and SS_GP used in place of the usual variables from the standard wind system. This gives a wind measurement that is not dependent on any information from the standard system except temperature.

The remaining aspect to discuss in this report is the performance of the gust pod at high rate. In Section 5, that discussion will be combined with consideration of the standard wind measurements and those as modified by using the pitot-static sensor.

5 High-Rate Measurements

There are now four ways to determine the wind vector:

1. Use the standard measurements from the radome, standard pitot tube, ambient pressure, and velocity relative to the ground. This is the system discussed extensively by Cooper et al. [2016];
2. Modify that system by using the airspeed measurement from the new pitot-static system;
3. Calculate the wind from the measurements provided by the gust pod; and
4. Use the Laser Air Motion Sensor. This instrument is not part of the SOCRATES instrumentation so it will not be discussed further here, but it provided valuable confirmation that the corrections applied to the standard measurements of ambient and dynamic pressure remain valid as originally determined from the LAMS.

The emphasis in this section will be on the high-rate measurements.

5.1 Constructing the high-rate data.frame

In this report, to date only the high-rate measurements from the test and ferry flights and from research flight #1 are available to the author. The following discussion will be based mostly on the high-rate data from research flight #1. For the purposes of this study, those measurements were processed as discussed in Section 3 to incorporate the pitot-static sensor into the wind measurements (as new variables by using a variable TASTC that is derived from the revised dynamic pressure QCTC, the filtered ambient pressure measurement PSFCF, and the air temperature ATX. In other regards the wind calculation is standard and results in the variables WDTC, WSTC, WITC, UXTC and VYTC (analogous to the standard variables WDC, WSC, WIC, UXC and VYC). In the base of the gust pod, the algorithm developed in Section 4 leads similarly to variables WDG, WSG, WIG, UXG and VYG. A processor for adding these variables to the original netCDF file is discussed in the final section, but for use in this report all

those variables have been added to a standard high-rate data.frame. The relevant code is in the “getHRdata” chunk of R code in this program file.

Figures 16 and 17 show comparisons among the three measurements of the wind vector. All are in very good agreement, although there are occasional spikes in all that will be a problem for calculations of the spectral variance.

5.2 Removing spikes

Spikes are a prevalent problem in the high-rate data from SOCRATES flight #1, and they will add significant noise to the measured variance spectra, so their removal is an important requirement before the spectra are constructed. The method used is in “removeSpikes.R” that is soon to be added to the Ranadu package but for now is a separate file in the “Ranadu/NewWork” directory. The algorithm used is to calculate a rolling mean and rolling standard deviation for the supplied variable, set any measurements that depart from the rolling mean by more than “sdLimit” standard deviations (default value is sdLimit=4) as missing, and then interpolate to fill in missing values. This removes most of the visibly obvious spikes, but whatever causes the spikes may also produce lower-amplitude fluctuations that are not removed by this algorithm, so it is important to determine the source of the spikes. This is unresolved at present (at least to the author’s knowledge).

5.3 Lateral variance spectra

Figure 18 shows the three measurements of the spectral variance¹¹ of the lateral horizontal wind component “VY”. All three are reasonable and VYC and VYTC (the standard variable and that based on the pitot-static airspeed) are consistent, but there appears to be some deficit in the spectral variance for VYG around 0.3 Hz.¹² All are in reasonable agreement above 1 Hz. For the vertical wind (also a lateral component and so expected to show the same spectral variance as VY), the spectra are shown in Fig.19. The spectrum from the gust pod (variable WIG) is significantly below that from the other two measurements for frequencies about about 0.3 Hz and is significantly different from the spectral variance of VY in this region, so it appears to be suspect. The airflow distortions may suppress response to vertical wind at high frequency. This result differs from that in project DEEPWAVE, where the spectral variance of the vertical wind appeared to be very good and consistent with the spectrum of VY. The major difference in aircraft configuration is the presence of the large pods in SOCRATES, so it may be that the resulting airflow distortion interferes with the ability of the gust pod to measure high-frequency components of the vertical wind. The other two measurements of vertical wind (WIY and

¹¹The orange lines indicate the expected -2/3 slope for the frequency-weighted spectrum, with factor-of-ten difference in the corresponding eddy dissipate rate between successive lines. The heavy dotted line corresponds to an eddy dissipation rate of $10^{-4} \text{ m}^2 \text{ s}^{-3}$. In plots of the longitudinal spectrum (as below for UX), the reference lines are adjusted for the expected 4/3 ratio between longitudinal and lateral spectra.

¹²A conjecture is that, because this is a frequency where the aircraft responds strongly to changes in airflow, there might be some residual effect on airflow distortion in this frequency range that dampens the response to real sideslip fluctuations.

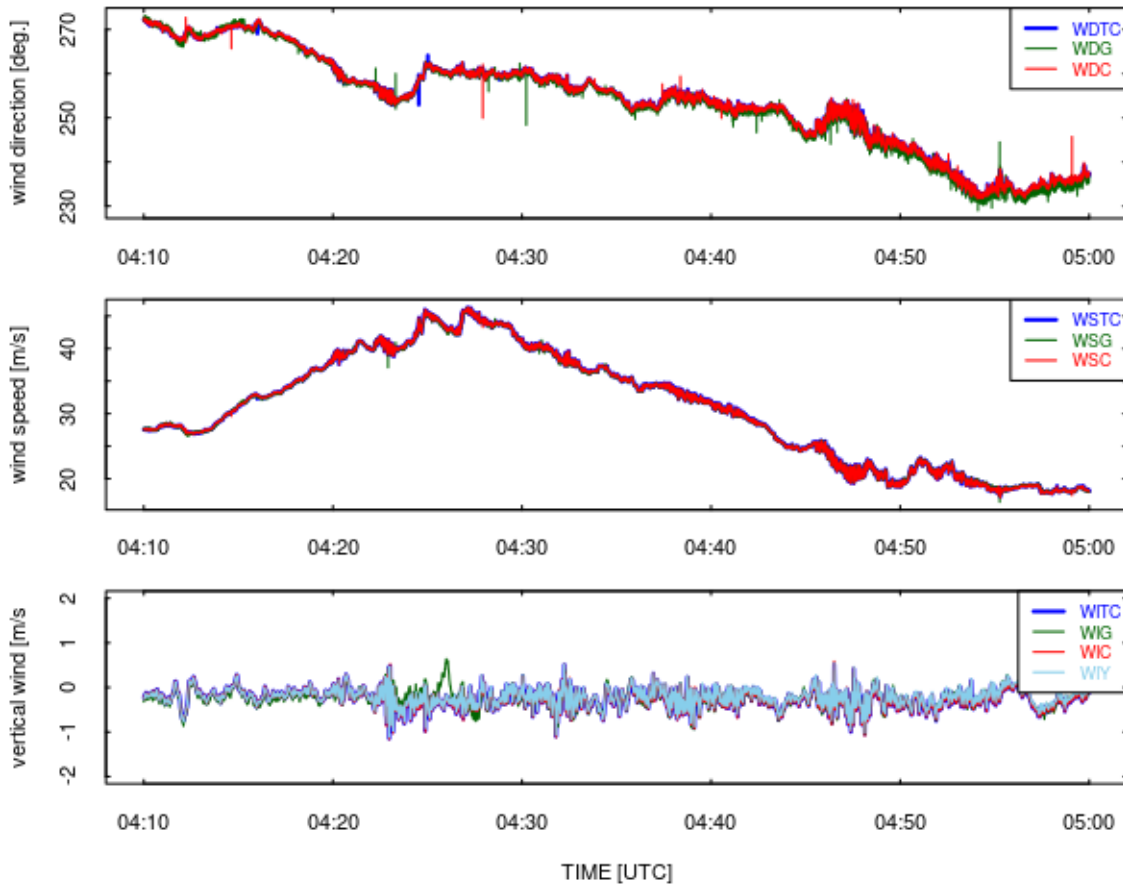


Figure 16: Comparisons of the wind components from the standard system (suffix C), the standard system modified to use the pitot-static sensor for airspeed (suffix TC), and the gust pod (suffix G). The vertical wind measurements have a 60-s smoothing function applied because otherwise the noise obscures the differences, and for vertical wind the added variable WIY (cyan color) is the vertical wind that uses the angle-of-attack from the complementary-filter algorithm. Data from high-rate processing for SOCRATES research flight 1, 4:10:00 – 5:00:00 UTC.

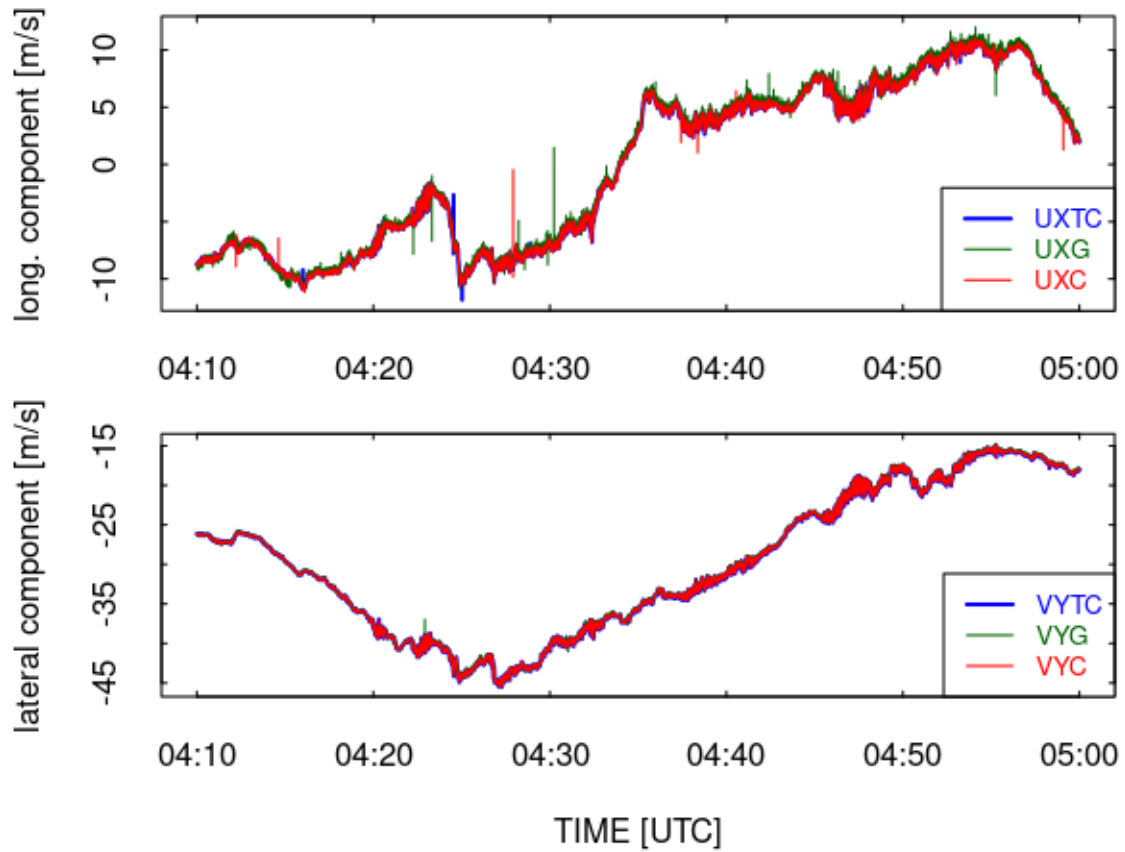


Figure 17: The lateral (VY) and longitudinal (UX) components of the horizontal wind for the same period shown in the previous figure.

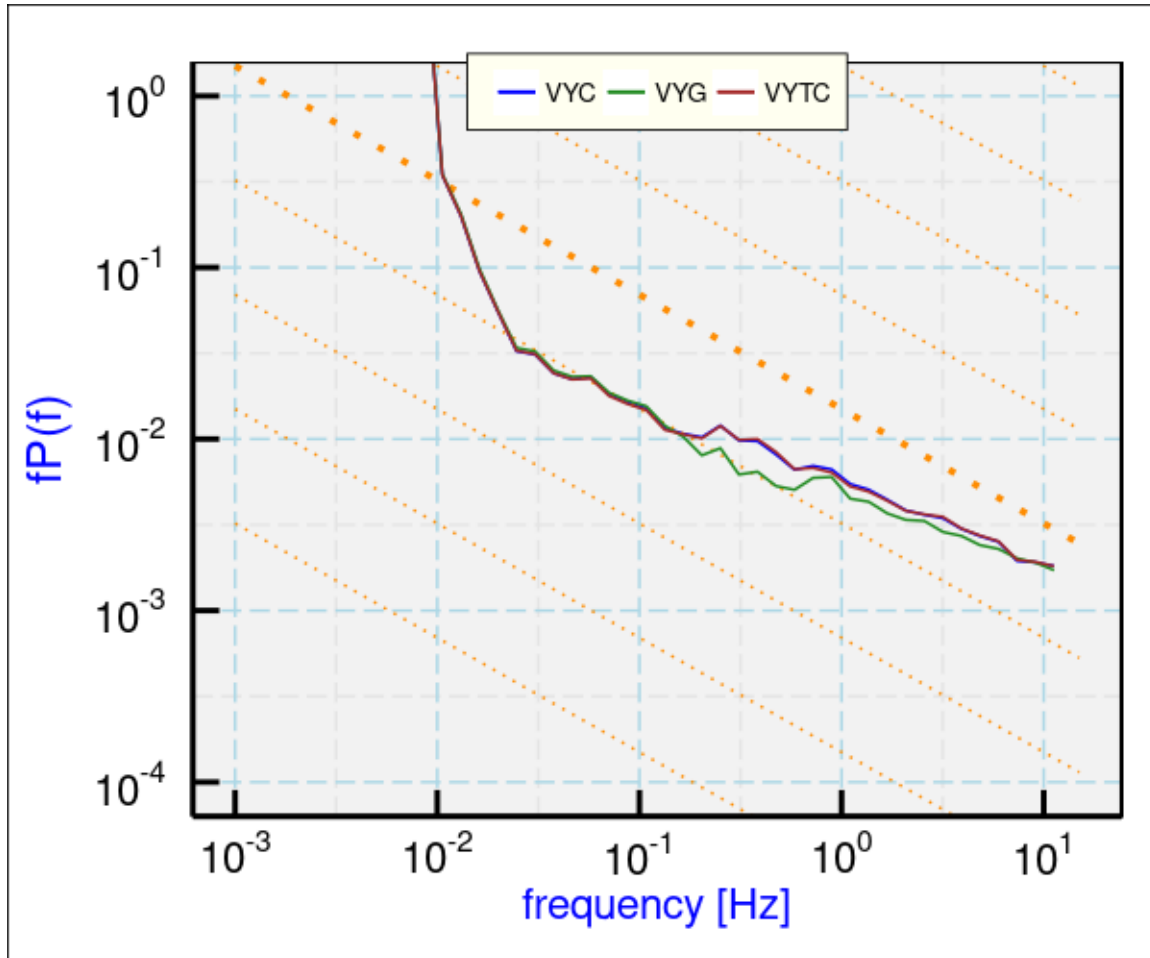


Figure 18: Variance spectra for measurements of the lateral horizontal wind from SOCRATES flight 1, 4:10:00 – 5:00:00, a region with some mild turbulence.

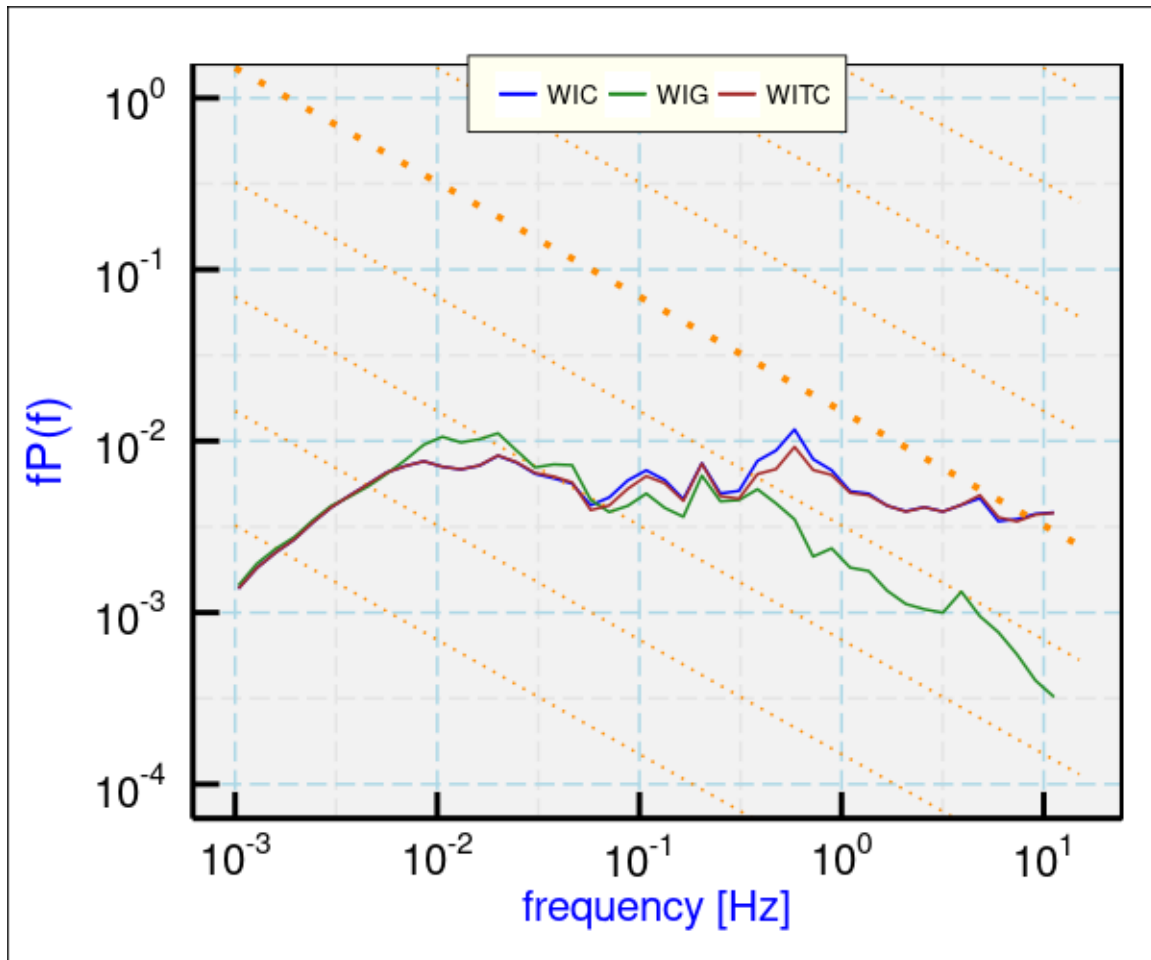


Figure 19: Variance spectra for measurements of the vertical wind from SOCRATES flight 1, for the same region as the previous figure.

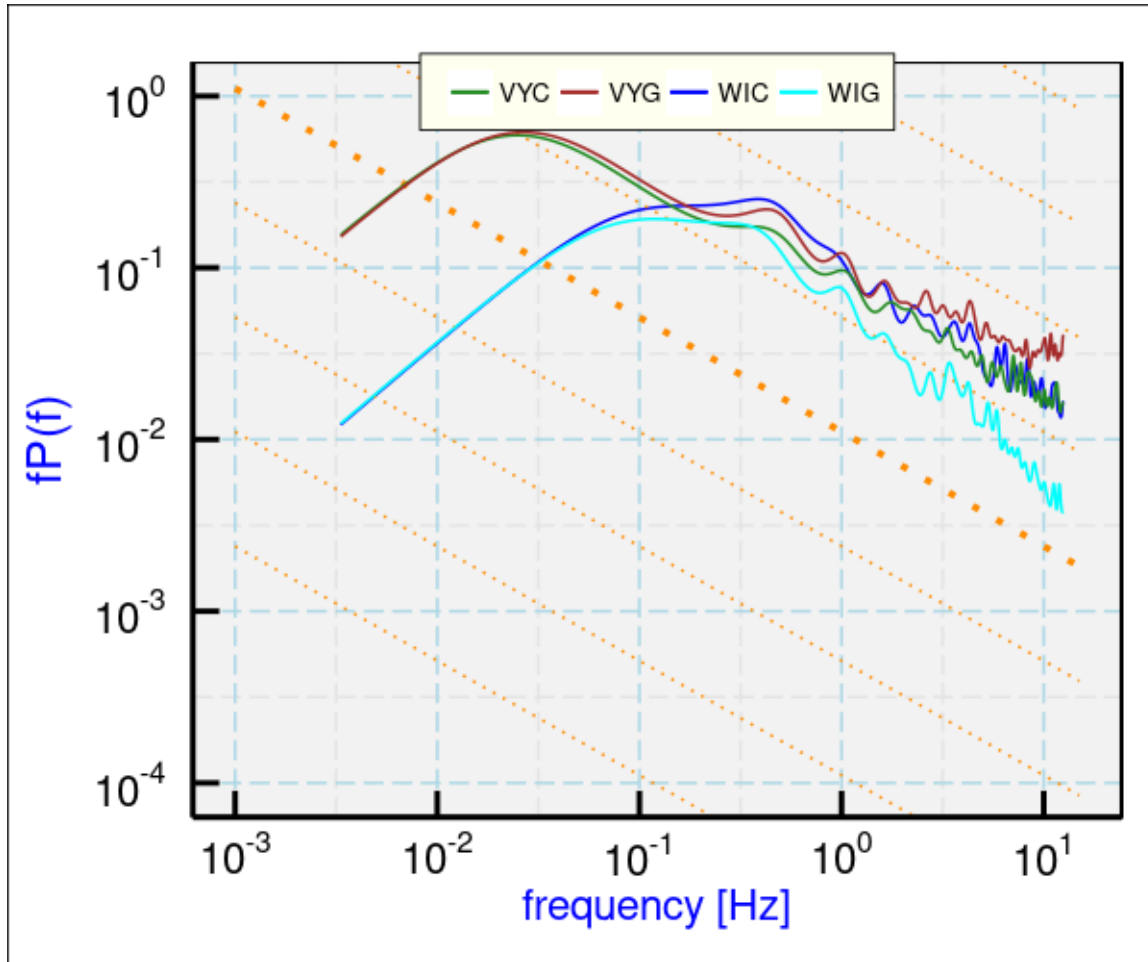


Figure 20: Variance spectra for lateral wind components (WI=vertical and VY=starboard) for the period from 4:56:00 – 5:01:00 UTC on SOCRATES flight 8.

WITC) appear to be too high for frequencies above about 3 Hz, and this apparent problem is still greater without spike removal. The spectrum for WIC is the same. This is an unfortunate weakness in the measurements, and this extended effort to use the new measurements has not produced an improvement in the vertical wind. The departure from expected shape above 3 Hz does not appear to be a simple random-noise spectrum, which would be expected to increase linearly with frequency in this plot, so it is unclear what causes the departure from expected shape.

The turbulence is relatively mild in this region, with representative eddy dissipation rate of about $10^{-5} \text{ m}^2 \text{ s}^{-3}$. More turbulent regions lead to spectra with different characteristics and different problems. A good example is Flight #8, 4:56:00 – 5:01:00, where the estimated eddy dissipation rate is about $2 \times 10^{-3} \text{ m}^2 \text{ s}^{-3}$. In this case, both standard lateral spectra (for WIC and VYC, blue and green lines in Fig. 20) appear very good and consistent with each other, without the suspicious flattening evident in Fig. 19. Also in other cases examined in SOCRATES, these lateral variance spectra show the expected high-frequency slope and there are no evident problems. Variance spectra for other related variables WIY, WITC, VYTC are not distinguishable

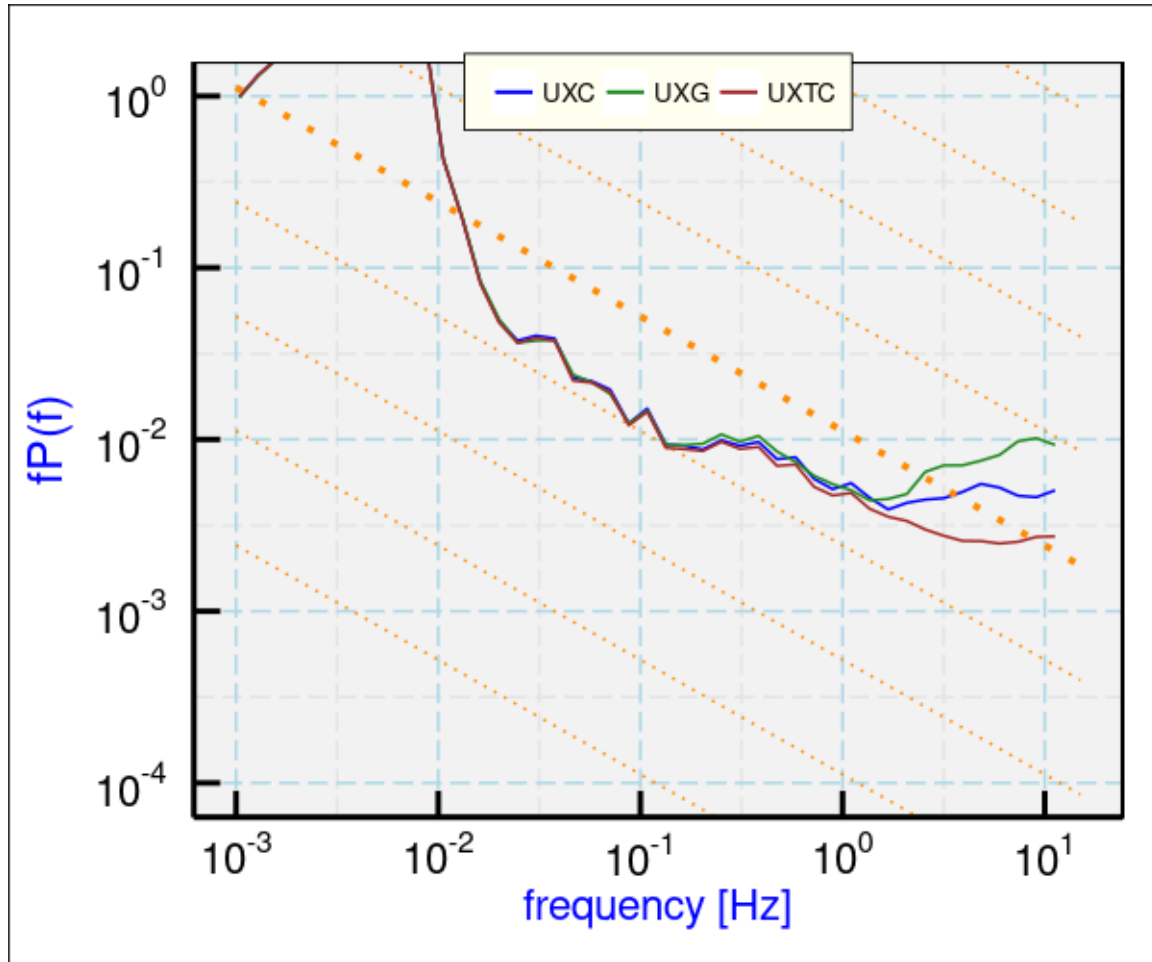


Figure 21: Variance spectra similar to those shown in the previous two plots and for the same time interval, but for the longitudinal spectrum UX.

from those for WIC and VYC. However, the gust-pod measurements continue to have problems at high frequency. The spectrum for the vertical wind (WIG, brown line) is lower than the other measurements, has apparent low intensity at high frequency, and is generally not in agreement with other measurements for frequencies above about 0.3 Hz. Effects of flow distortion around the aircraft apparently affect the ability of this instrument to measure high-frequency fluctuations in the vertical wind. The gust-pod measurement of lateral wind, VYG, looks better but too high for frequencies above about 3 Hz.

5.4 Longitudinal variance spectra

Similar plots of spectral variance for the longitudinal components (UX) are shown in Fig. 21, for same time period for which the lateral spectra were shown in Fig. 18. In this case (with moderately low turbulence), the gust-pod measurements appear to be biased upward with a

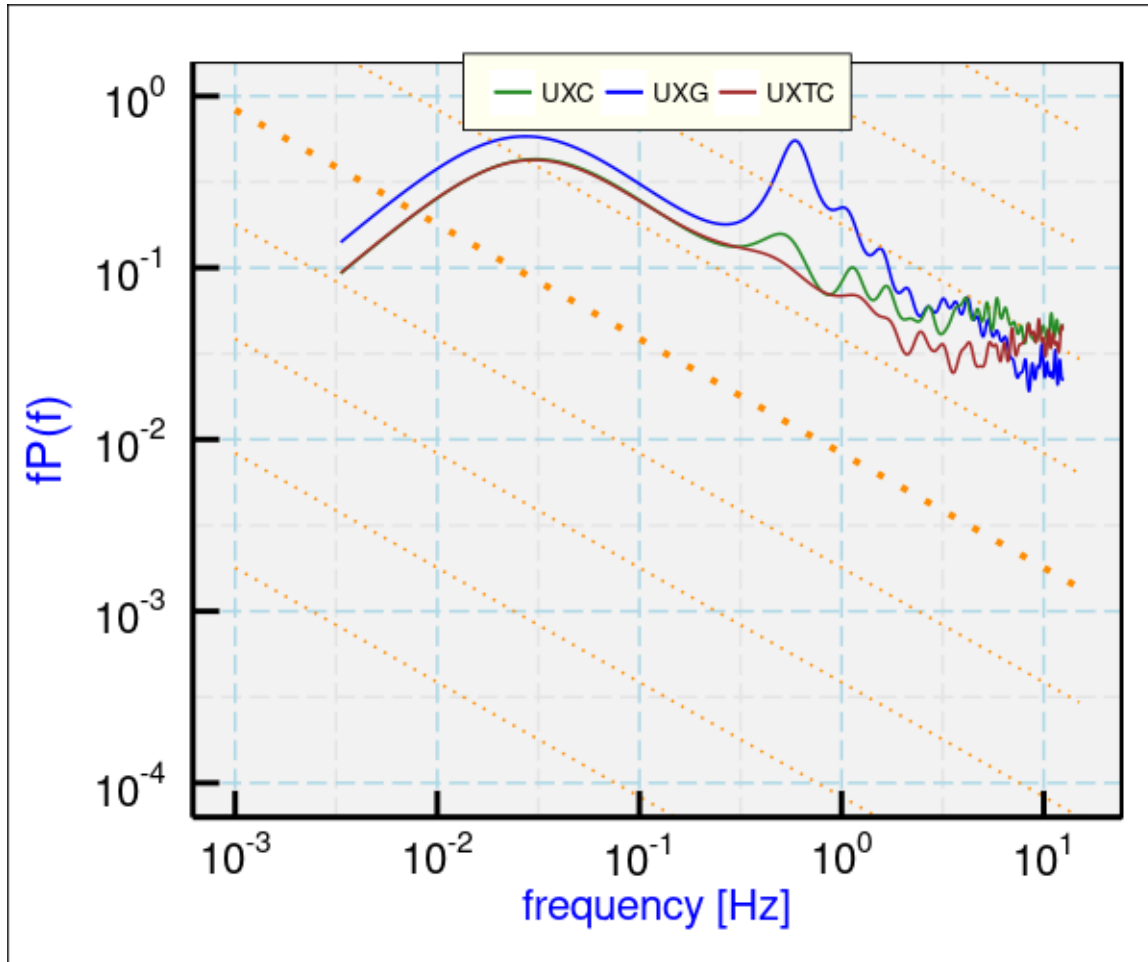


Figure 22: Variance spectra for a region of more intense turbulence, for the longitudinal wind component (UX). UXC is the standard measurement, UXTC is based on the airspeed measured by the pitot-static sensor, and UXG is from the gust pod. Data from SOCRATES research flight 8, 4:56:00 – 5:01:00 UTC.

high-frequency slope characteristic of noise for frequencies above about 1 Hz.¹³ For this measurement, the pitot-static measurement of airspeed has produced a significantly better spectral variance for the frequency range above 1 Hz, but even that spectrum shows some flattening above about 4Hz. This may be a residual effect of whatever is causing the frequent spikes in the measurements from this flight or it may result from high-frequency flow-distortion effects in the airflow approaching the pitot-static sensor, which is close to the aircraft skin.

For the case with high-intensity turbulence, for which lateral spectra were shown in Fig. 20, Fig. 22 shows the variance spectrum for measurements of the longitudinal component. The standard measurement (UXC, green line) produces estimates of spectral density that depart from the expected slope above about 1 Hz, reflecting the problem arising from sample-line resonance

¹³The calculation of airspeed used the same decomposition into fast and slow components as for angles of attack and sideslip, except with a cutoff frequency at 1/60 Hz to try to improve the high-frequency response, but that did not improve the variance spectra relative to a fit that did not use this decomposition.

discussed before. The new measurement from the pitot-static sensor (UXTC) looks much better but still shows indications of noise beginning around 5 Hz. For the gust pod (blue line), two features are notable, the 0.5-Hz peak and the apparently good performance near the Nyquist limit. The former is probably an erroneous effect of the flow-distortion around the aircraft and some coupling of that to natural oscillations of the aircraft in pitch and heading. The latter effect suggests that, if it becomes crucial to some aspect of research, it may be feasible to “blend” the high-frequency measurements from the gust pod with the lower-frequency components of the pitot-static-based measurement of longitudinal wind.

The longitudinal spectra would look much worse without the spike-removal procedure discussed in the preceding section, so some of the remaining problem may arise from smaller spikes not removed by that algorithm. This is another indication that it is important to find and correct the source of those spikes, which have been prevalent during SOCRATES.

6 Summary and Conclusions

The NCAR Technical Note TN-528+STR (Cooper et al. [2016]) indicated that the estimated standard uncertainty for measurements of wind from the NSF/NCAR GV research aircraft are about 0.12 m/s for the vertical wind and 0.4 m/s for each component of the vertical wind, but it also discussed a problem with measurements of the variance spectrum of the longitudinal component of the wind (i.e., along the aircraft longitudinal axis) at frequencies above about 1 Hz. The present report extends that study in these ways:

1. Measurements from the LAMS were used to confirm that the corrections being applied to measurements of ambient and dynamic pressure remained valid in Feb. 2017, at the time of the ARISTO-2017 campaign.
2. Prior to Feb. 2017, a new pitot-static sensor was installed on the GV, largely as a result of Dick Friesen’s initiative. The purpose was to reduce the contamination of the longitudinal-wind measurement that results from resonance in the sample lines. That sensor was recorded in SOCRATES, so in this report a processing scheme was developed to use with that sensor. New measurements of the wind were then calculated using the revised airspeed from that sensor. The conclusion of an evaluation of those wind measurements is that:
 - (a) At low rate (<0.5 Hz) the measurement of airspeed and hence of wind is consistent with the standard system. This result did not rely on any calibration to match the standard system because the pitot-static tube provides an independent measurement of the total pressure (ambient plus dynamic) that agrees with the standard system without calibration.
 - (b) At high rate (>0.5 Hz) the variance spectrum of the airspeed or the longitudinal component of the wind is in better agreement with the expected $-5/3$ slope up to about 5 Hz, but above that frequency some noise is still present in the spectrum. The source of this residual noise has not been determined. Despite that deficiency,

the longitudinal variance spectrum has been improved significantly by the addition of the pitot-static tube, and future data processing should be based on the dynamic pressure from that sensor.

- (c) A separate memo (attached) describes how the measurements from the pitot-static sensor can be incorporated into the standard processing for the GV. A processing scheme to add the new measurements based on the pitot-static sensor has been added to the QAtools shiny app. For low-rate measurements the new algorithm is trivially simple and can be incorporated immediately into standard processing. At high rate, it is helpful to apply some filtering to one of the variables used, so that may be done best in post-processing for now, although the low-rate algorithm could also be used and would usually produce an improvement even at high rate.
3. The gust pod (based on an 858 probe mounted in a canister under the GV wing) was flown in SOCRATES for the first time in conjunction with the large underwing pods. They apparently changed the airflow distortion ahead of the gust pod so that the earlier processing scheme led to some offsets in the measurements (for example changing the airspeed calculation by 2 m/s). Therefore a new procedure was developed for using the gust pod in this configuration. That procedure then was used to calculate wind measurements from the gust pod that are mostly independent of those from the radome-based system. These are the significant results from study of those measurements of wind from the gust pod:
- (a) At low rate, measurements of airspeed and of the angles of attack and sideslip were consistent with measurements from the standard system, so the gust pod provides a useful all-weather backup to that system.
 - (b) The variance spectra for measurements from the gust pod showed various problems that appear associated with airflow distortion. There appears to be possible contamination of the longitudinal-wind spectrum at frequencies expected for phugoidal or Dutch-roll aerodynamic instabilities. These are removed by the standard measuring system, but the distortion of airflow around the aircraft may be delayed in ways that could affect the airflow at the gust pod. Although this is a conjecture, there is a clear problem at around 5 Hz in the longitudinal-wind spectrum from the gust pod. The spectral density for the vertical-wind seems significantly too low at high frequency. The suggested conclusion is that measurements from the gust pod do not lead to useful estimates of spectral variance.
 - (c) A processing scheme was developed to produce measurements of the wind vector and the airspeed from the variables that the gust pod measures. It appears to be too complex to be worth incorporating into the standard processor “nimbus” because, to get useful measurements, it was necessary to use various transformations among coordinate systems and fitting to measurements that are split into complementary low-frequency and high-frequency components. A processor to add the wind measurements from the gust pod to the standard netCDF files has been incorporated into the QAtools shiny app.
4. In the 25-Hz netCDF files produced during the field project, there are spikes in many of the measurements that do not appear to be real and that, unless removed, will contaminate

the variance spectra for those measurements. For the plots of spectral variance shown in Sect. 5 of this report a spike-removal algorithm was used to reduce the effects of such spikes, but other spikes of lower amplitude may remain and contaminate the spectra. Follow-up investigation is needed to determine the source of these spikes.

7 The Processor

The transformations from the gust-pod reference frame to the aircraft reference frame are involved and likely to need careful check-out when implemented, so instead the recommended initial approach is to use a separate processor to add these variables to the netCDF files. This processor is included in the QAtools shiny app, under the “Special” tab, as an option in the “Options for Wind” tab. That tab also supports addition of wind variables based on the pitot-static sensor, the complementary-filter-based angle of attack and associated vertical wind as well as the addition of a high-pass-filtered version of the vertical wind. A new file with the suffix “Y” added to the name (e.g., SOCRATESrf01Y.nc) is created using a copy of the original, and these variables are added to that file:

New Variable	Description	Reference
AKY	angle-of-attack, complementary-filter algorithm	Eqns._(1), (2)
WIY	vertical wind, based on AKY	Sect. 4.4
ROC	rate of climb of the aircraft, using hydrostatic equation	Cooper [2017] (Sect. 3.1)
AK_GP	angle-of-attack from the gust pod, in the a-frame	Sect. 4.3
SS_GP	sideslip angle from the gust pod, in the a-frame	Eqn. (3)
WIG	vertical wind calculated using AK_GP and SS_GP	
WDG	wind direction using AK_GP and SS_GP	
WSG	wind direction using AK_GP and SS_GP	
TASG	airspeed calculate from the gust-pod	Eqn. (4)
UXG	wind from the gust pod, longitudinal component	Eqn. (3.1)
VYG	wind from the gust pod, lateral (starboard) component	
TASTC	airspeed calculated from the pitot-static sensor	Eqn. (3.1)
WDTC	wind direction based on the pitot-static sensor	
WSTC	wind speed based on the pitot-static sensor	
WITC	vertical wind based on the pitot-static sensor	
UXG	longitudinal wind based on the gust pod	
VYG	lateral wind based on the gust pod	
UXTC	longitudinal wind component based on TASTC	
VYTC	lateral (starboard) wind component based on TASTC	

Instructions for running this special processor are included in the QAtools user guide. See the discussion of the “Special” tab. The code used for this processing is available as part of the GitHub archive for this report; see “WindInSocrates.Rnw” and the associated chunk “Ad-dWind.R”.

A Reproducibility

This document is constructed in ways that support duplication of the study. The code that generates the plots and implements the processing algorithm is incorporated into the same file that generated this document via \LaTeX , using principles and techniques described by Xie [2013] as implemented in the R package 'knitr' (Xie [2014]). The program, 'WindInSOCRATES.Rnw', is archived on 'GitHub' in the directory at this URL. There is some supplemental material in that directory, including the workflow document, the bibliography and some code segments saved in the "chunks" subdirectory. This full directory should be downloaded in order to run the program. The calculations use the programming language R (R Core Team [2016]) and were run within RStudio (RStudio [2009]), so this is the most straightforward way to replicate the calculations and the generation of this document.

A package named Ranadu, containing ancillary functions, is used extensively in the R code. It is available on GitHub as <https://github.com/WilliamCooper/Ranadu.git>. The version used for calculations in this technical note is included in the 'zip' archive listed below.

The original files containing the data as produced by the NCAR Earth Observing Laboratory, Research Aviation Facility, were in netCDF format (cf. this URL), but particularly for SOCRATES the files used are preliminary data not yet released for general use. The officially released data files should be used once they are available. The subset data frames constructed from those preliminary netCDF files are saved and can be provided by the author, if or when that is consistent with the project data policies. The code in the GitHub archive has appropriate 'load' commands to read these data files from a subdirectory named 'Data' (/Data or ~/Data or /home/Data) but this is not part of the GitHub repository because it is too large to be appropriate there. To reproduce this research, those data files have to be transferred separately from the NCAR HPSS to the 'Data' directory. Some use has been made of attributes assigned to the data.frames and the variables in those data.frames. All the attributes from the original netCDF files have been transferred, so there is a record of how the original data were processed, for example recording calibration coefficients and processing chains for the variables. Once the data.frames are loaded into R, these attributes can be viewed and provide additional documentation of what data were used. Key information like the processing date, the program version that produced the archive, and the selection of primary variables for various measurements thus is preserved.

(See the related list of project components on the next page.)

PROJECT: WindInSOCRATES
ARCHIVE PACKAGE: WindInSOCRATES.zip
CONTAINS: attachment list below
PROGRAM: WindInSOCRATES.Rnw
SPECIAL DATA FILES: AKRDdata.Rdata
WORKFLOW DOCUMENT: WorkflowWindInSOCRATES.pdf
GIT: <https://github.com/WilliamCooper/SocratesQA.git>

Attachments: WindInSOCRATES.Rnw
WindInSOCRATES.pdf
NoteReSOCRATESwindProcessing.pdf
WorkflowWindInSOCRATES.pdf
WAC.bib
chunks/*
SessionInfo

References

- W. A. Cooper. A Kalman filter to improve measurements of wind from NSF/NCAR research aircraft. NCAR technical note NCAR/TN-540+STR, Earth Observing Laboratory, NCAR, Boulder, CO, USA, sep 2017. URL <http://dx.doi.org/10.5065/D61N7ZTS>.
- W. A. Cooper, S. M. Spuler, M. Spowart, D. H. Lenschow, and R. B. Friesen. Calibrating airborne measurements of airspeed, pressure and temperature using a doppler laser air-motion sensor. *Atmospheric Measurement Techniques*, 7(9):3215–3231, 2014. doi: 10.5194/amt-7-3215-2014. URL <http://www.atmos-meas-tech.net/7/3215/2014/>.
- W. A. Cooper, R. B. Friesen, M. Hayman, J. B. Jensen, D. H. Lenschow, P. A. Romashkin, A. J. Schanot, S. M. Spuler, J. L. Stith, and C. Wolff. Characterization of uncertainty in measurements of wind from the NSF/NCAR Gulfstream V research aircraft. NCAR technical note NCAR/TN-528+STR, Earth Observing Laboratory, NCAR, Boulder, CO, USA, jul 2016. URL <http://n2t.net/ark:/85065/d7qr4zqr>.
- R Core Team. *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria, 2016. URL <http://www.R-project.org>.
- RStudio. *RStudio: Integrated development environment for R (Version 0.98.879)*, 2009. URL <http://www.rstudio.org>.
- Y. Xie. *Dynamic Documents with R and knitr*. Chapman and Hall/CRC, Boca Raton, Florida, 2013. URL <http://yihui.name/knitr/>. ISBN 978-1482203530.
- Y. Xie. *knitr: A general-purpose package for dynamic report generation in R*, 2014. URL <http://yihui.name/knitr/>. R package version 1.6.