Examination of a simple pulse-blanking technique for radio frequency interference mitigation

Noppasin Niamsuwan and Joel T. Johnson
ElectroScience Laboratory, Department of Electrical and Computer Engineering, Ohio State University, Columbus, Ohio, USA

Steven W. Ellingson
Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, USA

Received 18 August 2004; revised 30 November 2004; accepted 12 January 2005; published 8 June 2005.

[1] Radiometry at L band can be adversely impacted by radio frequency interference (RFI) due to the presence of numerous sources, especially pulsed RFI from radars operating below 1400 MHz. RFI mitigation is very important to deal with this problem. A simple strategy for reducing pulsed RFI, termed “asynchronous pulse blanking” (APB), has been implemented in a digital receiver developed at Ohio State University. This paper presents results from a simulation of the APB algorithm. Several aspects of algorithm use and performance are reported, including means for choosing the algorithm’s parameters and the robustness of the method in a realistic RFI environment. Effects of the blanking process on the final output are also examined.


1. Introduction

[2] Radio frequency interference (RFI) mitigation is very important for radiometry at L band due to the presence of numerous interference sources, including strong pulsed RFI from ground-based aviation radars (GBARs) [Ellingson, 2003; Ellingson et al., 2003a]. A number of mitigation techniques have been proposed. Friedman [1996], Weber et al. [1997], and Leshem et al. [2000] describe different methods to detect and reject impulsive interference. For radiometers operating at a sufficiently high temporal sample rate, a simple strategy for reducing pulsed RFI is to remove incoming data whose power exceeds the mean power by a specified number of standard deviations. It may also be advantageous to remove data within a specified time region before and after this “trigger” data, to ensure that any predetection and postdetection “pulse” information is successfully removed. Such an algorithm has been implemented in a digital receiver-based radiometer developed at Ohio State University ElectroScience Laboratory [Ellingson et al., 2003b; OSU IIP project document server, 2004, available at http://esl.eng.ohio-state.edu/~swe/iip/docserv.html]; the process is termed “asynchronous pulse blanking” (APB) because no periodic properties of the interference source are assumed. Although successful performance of this algorithm has been qualitatively demonstrated through local experiments with the digital radiometer [Hampson et al., 2004], a detailed quantitative assessment of its performance in a variety of RFI environments has not been reported. In addition, while the algorithm is conceptually simple, developing a process which is both automatic and robust requires careful consideration of the implementation details.

[3] To address this issue, a simulation study of the APB algorithm was initiated using data obtained from the L-Band Interference Surveyor/Analyzer (LISA), an airborne instrument developed for observing the RFI in the region 1200–1700 MHz [Ellingson et al., 2003a; Ellingson and Johnson, 2003; Johnson and Ellingson, 2003]. During January–February 2003, LISA was deployed on NASA’s P3-B aircraft to observe RFI in flights in the US and Japan. This data set is very useful for assessing the APB algorithm, since many RFI environments were observed that include multiple sources of interference.
The next section provides a basic description of the digital receiver–based radiometer, and a detailed summary of the APB algorithm follows in section 3. Information on the LISA instrument and data set is then provided in section 4, and simulation results presented in section 5.

2. Interference-Suppressing Microwave Radiometer

Under the support of the NASA “Instrument Incubator Program” (IIP), development of a digital receiver–based L-band radiometer was initiated in December 2001. Although the project is focused on passive microwave observation of the Earth for L-band remote sensing, the radiometer can be applied for L-band radio astronomy as well, and both applications are severely impacted by RFI.

A block diagram of the radiometer is shown in Figure 1. The analog front end downconverts an 80 MHz swath of spectrum from L band to 150 MHz, and samples this signal at 200 megasamples per second (MSPS) using 10 bits. Note the system can also be operated at frequencies other than L band simply by modifying the analog front end and downconverter sections. Because the analog IF is in the second Nyquist zone of the A/D, the digital passband is centered at 50 MHz and is spectrally reversed. The “Digital IF” (DIF) FPGA module downconverts this to 0 Hz (so now the samples are complex valued), filters to 50 MHz bandwidth, decimates by 2, and then upconverts to a center frequency of +25 MHz (still complex). The data emerges from the DIF module in 16-bit “I”+16-bit “Q” format at 100 MSPS. The sample process is applied to a separate, independently tunable 50-MHz swath at L band, with the difference that the digital output is centered at ~25 MHz. The two 50 MHz bands are simply added together to form single 100 MHz bandwidth signal.

Following the DIF output is a cascade of FPGA modules which can be programmed to perform a variety of functions. Our favored strategy currently is as shown in Figure 1: mitigation of radar pulses using APB, channelization into 100-kHz bins using a 1K Fourier fast transform (FFT), frequency domain blanking, and integration to generate power spectra.

The APB is designed to detect and blank radar pulses, which typically are the dominant source of external L-band RFI below 1400 MHz. Radar pulses range from 2 to 400 ms in length and occur 1–75 ms apart [Ellingson, 2003]. To detect these pulses, the APB maintains a running estimate of the mean and variance of the sample magnitudes. Whenever a sample magnitude greater than a threshold number of standard deviations from the mean is detected, the APB blanks (sets to zero) a block of samples beginning from a predetermined period before the triggering sample, through and hopefully including any multipath components associated with the detected pulses. APB operating parameters are adjustable and can be set by the user.

Following the APB is a length-1K complex FFT, which achieves approximately 98% duty cycle in performing the FFT computations (i.e., 98% of the data is FFTed, and the rest is lost). A triangular window is applied before the FFT. Planned but not yet implemented is a frequency domain blanking module, which is similar in concept to the APB, except applied independently to...
each frequency bin. The purpose of this module will be to exploit the processing gain achieved through channelization to detect and excise weak, relatively narrowband RFI. The FFT output is processed through a “spectral domain processor” (SDP) module which computes magnitude squared for each frequency bin and computes a linear power average over many FFT outputs. These results are passed at a relatively low rate to a PC via a capture board. Total power can be computed by summation of frequency bins within the digital hardware, or the same process can be implemented within the PC for increased flexibility in monitoring RFI, selecting subbands, and so on.

[10] This radiometer has been used in both radio astronomy and remote sensing experiments locally at the ElectroScience Laboratory [Hampson et al., 2004] and at the Arecibo observatory in Puerto Rico [Ellingson and Hampson, 2003a, 2003b]. The results qualitatively show the APB approach to be successful for pulsed RFI mitigation, although a detailed performance assessment has not yet been achieved.

3. APB Algorithm

3.1. Basic Concept

[11] The APB algorithm is intentionally simple to allow implementation in hardware (FPGAs). For each sample $x$ in the data stream, $|x|^2$ (the squared magnitude of the sample) is computed and compared to a threshold level $\delta$. Detection is declared if $|x|^2$ exceeds this threshold. The threshold level is defined to be $\beta \sigma + m$, where $\sigma$ and $m$ are the standard deviation and mean of $|x|^2$, respectively. Both the mean ($m$) and standard deviation ($\sigma$) are estimated from the data, as will be described in section 3.2.2. The parameter $\beta$ sets the “aggressiveness” of the blanker; for small $\beta$, the blanker may trigger on larger noise peaks, whereas a high value may allow weak pulses to pass through. Consequently, the $\beta$ parameter should be optimized to tradeoff loss of noise power integration time versus blanking performance.

[12] Because “pulses” detected by the algorithm typically begin before a detection is obtained, it is useful for the APB algorithm to have the capability of blanking information prior to a detection. To blank predetection samples, a buffer (memory) is required in the system. The detection operation is then performed on samples entering the buffer, while the blanking operation and all successive operations are performed on samples exiting the buffer. A simple first-in-first-out (FIFO) buffer suffices, and the length of this FIFO (FIFO_LENGTH) determines the maximum period that can be blanked before the detected sample.

[13] Even though the APB processor is capable of blanking all the samples in the buffer when a pulse is detected, this is not always necessary. Blankaging the whole buffer may remove desired noise information and should be avoided if possible. To control the length of the prepulse-blanking region, the parameter $N_{\text{wait}}$ is introduced. When detection is asserted, a process is initiated to wait for $N_{\text{wait}}$ samples before the FIFO output is set to zero. The number of predetection samples remaining in the buffer after $N_{\text{wait}}$ clock cycles have elapsed is then FIFO_LENGTH-$N_{\text{wait}}$. The FIFO output is then continually forced to zero for a specified number of clock cycles. The length of this blanking “window” is determined by the final APB parameter, $N_{\text{blank}}$. The set of parameters $\beta$, $N_{\text{wait}}$, $N_{\text{blank}}$, and FIFO_LENGTH are the basic conceptual quantities that control the APB process.

[14] Figure 2 illustrates simulated incoming signal power, and includes simulated RFI pulses. The blanking regions in the plot clarify how the $N_{\text{blank}}$ and $N_{\text{wait}}$ parameters are defined. It is clear from Figure 2 that $N_{\text{blank}}$ should be chosen large enough to remove a complete pulse and any possible multipath contributions, while the combination FIFO_LENGTH-$N_{\text{wait}}$ should be chosen sufficiently large to remove predetection contributions. The studies of section 5 investigate methods for choosing these parameters.

3.2. Hardware Issues

3.2.1. FIFO LENGTH

[15] Ideally, the FIFO should be made as large as possible in order to allow the largest predetection period. However, hardware issues (i.e., size of the FPGA used) limit the maximum FIFO size which can be included. For the IIP radiometer and for the remainder of the simulations performed in this paper, FIFO_LENGTH was set to 1024 because of size limitations of the FPGA components used in the prototype system.

3.2.2. Running Filter

[16] The APB algorithm requires a running estimate of the mean ($m_i$) and variance ($\sigma_i^2$) of the incoming signal power. These estimates are computed as

$$m_i = \mu_{\text{mean}} \times m_{i-1} + (1 - \mu_{\text{mean}}) \times |x|^2$$

$$\sigma_i^2 = \mu_{\text{var}} \times \sigma_{i-1}^2 + (1 - \mu_{\text{var}}) \times (|x|^2 - m_i)^2$$
where $m_{t-1}$ and $\sigma_{t-1}^2$ are the mean and variance generated on the previous clock cycle, and $\mu_{\text{mean/\text{var}}}$ is an average of $N$ samples. The above computations are accomplished by groups of fixed-point multipliers, adders, and delay registers. In addition, a state machine controls these filters so that contributions from incoming postdetection data that are blanked are not included in computing $m_t$ and $\sigma_t^2$.

### 3.2.3. Large Bit Width

[17] Because of the squaring operation in computing $||x||^2$ (represented originally as 16 bit $I$ and 16 bit $Q$), a wide data path of 32 bits is required to preserve accuracy for this quantity. The estimate of $\sigma^2$ further computes the square of $||x||^2$, requiring 64 bits. The high dynamic range of the filter constants $\mu \approx 1$ and $(1 - \mu) \approx 0$ further increases the number of bits required in estimating the running mean and variance of $||x||^2$. Use of 12 bits ($\mu = 0.999756$) to capture both $\mu$ and $(1 - \mu)$ was found desirable in setting an appropriate averaging length for the filter. The resulting arithmetic operations grow to a maximum size of 76 bits, and severely limit the computation speed of the detector because of this large size.

### 3.2.4. Processor Speed

[18] Because of the large bit growth of the APB processor and speed limitations of the FPGA components currently used, the detection operation cannot be performed on 100% of the incoming data. The current implementation of the design can process only one in four samples (25%) of the data for detection. A decimation controller is introduced to sub-sample the input data stream to every 4 samples for detection operations. However, since radar pulses are typically longer than this subsampling period, they are still detectable even under this limitation. This subsampling operation refers only to samples analyzed for detection, the remainder of the system, including the blanking operation, still operates on 100% of the incoming data. Note an average of four successive samples could also be used in the decimation controller rather than the subsampling method implemented, and would likely result in improved detection performance. However, for simplicity in implementation the subsampling method was utilized.

### 3.2.5. Control of Blanking Process

[19] When a detection is obtained, a simple state machine is initiated to wait for $N_{\text{wait}}$ clock cycles, then blank the output of the FIFO for $N_{\text{blank}}$ clock cycles. The state machine controlling this process is called a “blanking timing register” (BTR). Although the BTR is simple to implement in hardware, it still occupies a nonnegligible portion of the available FPGA logic. Therefore it is not possible to have an unlimited number of BTRs in the system. Because a BTR is unavailable to begin a blanking process for $N_{\text{wait}} + N_{\text{blank}}$ clock cycles after it has been “triggered”, the possibility of having no BTRs available to blank a detected pulse must be considered if only a finite number of BTRs exist.

[20] A limitation in the number of BTR’s also raises another issue: a large pulsed interferer presumably could trigger all available BTRs on successive clock cycles, with the resulting blanking windows overlapped almost completely at the FIFO output. The ability of the APB to blank detected pulses would then be compromised for approximately $N_{\text{wait}} + N_{\text{blank}}$ clock cycles when the initial BTR would become available again. This problem motivates the introduction of a parameter $N_{\text{sep}}$, defined to be a number of clock cycles after a detection during which no further BTRs will be triggered. If $M$ BTRs are available, then setting

$$N_{\text{sep}} = \frac{1}{M} (N_{\text{wait}} + N_{\text{blank}})$$

will ensure that a BTR is always available for triggering once $N_{\text{sep}}$ has elapsed. Although $N_{\text{sep}}$ values larger than the above value would also satisfy this requirement, minimizing $N_{\text{sep}}$ is desirable to ensure maximal pulse detection.

[21] While introduction of $N_{\text{sep}}$, manages the use of a finite number of BTRs, it also may compromise the ability of the APB to remove detected pulses, if a detection occurs within $N_{\text{sep}}$ clock cycles of an earlier detection. Although a new BTR process will not be initiated for $N_{\text{sep}}$ clock cycles, the APB algorithm can still be designed so that all postdetection signal information within $N_{\text{sep}}$ clock cycles will be blanked. Referring to Figure 2, the postdetection blanking region contains $N_{\text{blank}} + N_{\text{wait}} - \text{FIFO\_LENGTH}$ samples for a fixed $N_{\text{blank}}$. Choosing

$$N_{\text{sep}} \leq N_{\text{blank}} + N_{\text{wait}} - \text{FIFO\_LENGTH}$$

will ensure that all signal information within $N_{\text{sep}}$ samples after a detection will be blanked.

[22] Combining equations (3) and (4) results in the condition

$$N_{\text{blank}} \geq \frac{M}{M - 1} \text{FIFO\_LENGTH} - N_{\text{wait}}$$

to ensure that all samples exceeding the specified threshold will be blanked. Although this equation appears to require that $N_{\text{blank}}$ increase with $\text{FIFO\_LENGTH}$, the associated $N_{\text{wait}}$ would also likely increase with $\text{FIFO\_LENGTH}$ so that $N_{\text{blank}}$ would not necessarily increase. The digital radiometer prototype at present
contains 4 BTRs; this number will be assumed in the simulations of section 5. For this case with FIFO_LENGTH = 1024, \( N_{\text{blank}} \) should be chosen \( \geq 1366 - N_{\text{wait}} \). Note that even though the choice of equation (5) ensures that all samples exceeding the threshold will be blanked, successive pulses occurring within the \( N_{\text{sep}} \) period of an initial pulse will still not have a complete blanking window surrounding all their contributions.

4. L-Band Interference Surveyor/Analyzer (LISA)

[23] In order to perform a more complete assessment of the APB algorithm, it is desirable to study performance in a complex and varying RFI environment. However, experiments with the radiometer prototype to date have occurred only in the local environment and at the Arecibo facility. Furthermore, it is desirable to perform these assessments in a software rather than hardware environment, so that quantitative information can be obtained as APB parameters are varied for a fixed specific data set. The data set from the LISA sensor provides an excellent resource for these studies.

[24] LISA, completed in September 2002, was developed as a means to observe sources of radio frequency interference (RFI) in the region of 1200–1700 MHz from an airborne platform [Ellingson et al., 2003a; Ellingson and Johnson, 2003; Johnson and Ellingson, 2003]. Physically, LISA comprises a small antenna/front-end unit (AFEU) and an equipment rack. The antenna unit consists of a nadir-facing cavity-backed planar spiral antenna with an integrated custom RF front end including filtering and calibration circuitry. The antenna has a very broad pattern (approximately \( \cos \theta \)) and is reasonably well matched over the span of the observations reported here. The antenna unit is connected to an equipment rack mounted in the aircraft cabin by a long and fairly lossy coaxial cable. Although the cable loss degraded the sensitivity of the instrument, the resulting gain profile was an important factor in preserving the linearity of the system while observing strong RFI waveforms, this consideration was paramount, but comes at the expense of the system’s ability to detect weak RFI.

[25] Inside the equipment rack, the signal is delivered to a custom-designed coherent sampling subsystem. This subsystem uses a direct conversion receiver to tune (under PC control) anywhere between 1200 MHz and 1700 MHz. “I” and “Q” signals at baseband are low-pass filtered with \( \sim 7 \) MHz cutoff and sampled at 20 MSPS, yielding a digitized bandwidth of \( \sim 14 \) MHz. The output samples are queued in a 16K sample–long first-in-first-out (FIFO), whose contents are acquired by means of the PC parallel port. These 16K “captures” represent an 819.2 \( \mu \)sec sample of the received field at the current center frequency, and require approximately 1 sec to be transferred and recorded on the computer. During flight, the coherent sampling system was successively tuned through center frequencies of 1250, 1264, \ldots, 1698 MHz; these distinct bands are called channel 1, 2, etc. in what follows. For each channel, 5 819.2 \( \mu \)sec subsequent captures are performed before tuning to the next center frequency. After completing a sweep of all channels, a backup operation was performed to a separate storage component in the computer system. The time required for this operation caused a 10 to 15 minute delay between “sweeps”. The final data set provides a high temporal resolution but very low duty cycle observation of 4.1 ms captured per LISA channel per 12 to 15 minutes.

[26] For the data sets considered here, LISA was installed in NASA’s P-3 research aircraft, which is based at the Wallops Flight Facility (WFF) located at Wallops Island, VA. The LISA antenna unit was mounted in the tail radome. The loss due to transmission through the radome is unknown and may be another factor degrading sensitivity. The data used in the simulations to follow were collected during a single flight on 3 January 2003 along an east-west track from Wallops Island, VA to Monterey, CA at approximately 6705 m. These data obviously include observation in a variety of RFI environments. Data from LISA channels 6–13 (1310–1428 MHz) is used in the simulations; for each of these channels, 145 captures (5 captures \( \times 29 \) sweeps) were available. LISA was also deployed in the “Wakasa Bay” remote sensing campaign (Colorado State University, Wakasa Bay field experiment description, available at http://rain.atmos.colostate.edu/Wakasa) in Japan through February 2003; data on the transit flights from Monterey to Japan were also collected [Johnson and Ellingson, 2003]. However, the RFI environments in these flights show less variation than that of the 3 January flight, and so these data are not considered further in this paper.

5. Simulations and Results

5.1. Choosing \( \beta^2, N_{\text{blank}}, \) and \( N_{\text{wait}} \)

[27] To investigate choice of \( \beta^2 \), data from LISA channel 7 was examined to determine the percent of samples exceeding a specified threshold determined by \( \beta \). Because the mean measured noise power can be expected to vary significantly throughout a cross-U.S. flight, the mean and variance estimates used to determine the threshold are computed by the running average described in section 3. This running average used \( N_{\text{blank}} = 2048 \) and \( N_{\text{wait}} = 0 \) in removing pulse contributions from the mean and variance computations, as described in section 3.2.2. Figure 3 illustrates the percent of samples out of the 145 16K captures that
exceed the threshold specified by $\beta^2$ on the horizontal axis. The dashed line included in Figure 3 represents the percent of exponentially distributed noise that would exceed the same threshold. An exponential, rather than Gaussian, noise distribution is more appropriate here because of the very short integration time of the incoming power for this system. In this case, the presence of RFI in the data set causes the percent blanked to exceed that of the simulated noise for large $\beta^2$ values; however as $\beta^2$ is reduced, the two curves become more identical because the detection of noise dominates both processes. In this case, the turning point with the LISA data appears to be in the range $\beta^2 \approx 30$ to 40. Further simulations with other LISA channels showed $\beta^2$ values ranging from 40 to 90 to be reasonable; smaller $\beta^2$ values clearly led to excessive blanking as should be expected.

[28] As mentioned previously, the parameter FIFO_LENGTH is fixed to 1024 (i.e., 51.2 \textmu sec of LISA data) in order to model the existing radiometer prototype. Because of this somewhat small size, tests using LISA channels 7–13 indicate that immediate blanking of the FIFO is preferable. For this reason, $N_{\text{wait}}$ is set to 0 in the remaining simulations. Future work will explore a preferred value of $N_{\text{wait}}$ for larger FIFO_LENGTH parameters.

[29] A simulation investigating the effect of $N_{\text{blank}}$ has also been performed. In this study, two threshold levels are defined: a higher “detection threshold” ($\beta^2 = 90$) used for pulse detection as usual while $N_{\text{blank}}$ is varied. The output of the blanker is then examined to determine the number of samples remaining that exceed a lower “reference threshold” ($\beta^2 = 30$ using the same mean and variance computations as those for the detection threshold). This quantity is labeled $P_{\text{out}}$ in what follows. The total number of samples exceeding the reference threshold in the original data is also determined, and labeled $P_{\text{in}}$. The ratio $(P_{\text{in}} - P_{\text{out}})/P_{\text{in}}$ then provides information on the effectiveness of the higher threshold blanker with a given $N_{\text{blank}}$ at removing lower-level pulse contributions. Figure 4 illustrates these quantities for a single LISA capture. Note in some cases, samples exceeding the higher threshold are not blanked in this data set; this is because such “trigger” samples may be missed by the subsampled detector if they are less than 4 samples long.

[30] Figure 5 plots $(P_{\text{in}} - P_{\text{out}})/P_{\text{in}}$ as a percentage (solid curve) from the entire channel 7 data set, with $N_{\text{blank}} \geq 1536$ which clearly satisfies equation (5). The curve shows only a modest variation with $N_{\text{blank}}$, and an $N_{\text{blank}}$ value in the range 1536 to 2048 appears appropriate in this case. This can also be interpreted as indicating that pulsed interference longer than 76.8 \textmu sec

![Figure 3. Percentage of samples exceeding threshold for given $\beta^2$. All 145 captures of LISA channel 7 data are used. See color version of this figure in the HTML.](image1)

$\beta^2$ values exceed the threshold specified by $\beta^2$ on the horizontal axis. The dashed line included in Figure 3 represents the percent of exponentially distributed noise that would exceed the same threshold. An exponential, rather than Gaussian, noise distribution is more appropriate here because of the very short integration time of the incoming power for this system. In this case, the presence of RFI in the data set causes the percent blanked to exceed that of the simulated noise for large $\beta^2$ values; however as $\beta^2$ is reduced, the two curves become more identical because the detection of noise dominates both processes. In this case, the turning point with the LISA data appears to be in the range $\beta^2 \approx 30$ to 40. Further simulations with other LISA channels showed $\beta^2$ values ranging from 40 to 90 to be reasonable; smaller $\beta^2$ values clearly led to excessive blanking as should be expected.

[28] As mentioned previously, the parameter FIFO_LENGTH is fixed to 1024 (i.e., 51.2 \textmu sec of LISA data) in order to model the existing radiometer prototype. Because of this somewhat small size, tests using LISA channels 7–13 indicate that immediate blanking of the FIFO is preferable. For this reason, $N_{\text{wait}}$ is set to 0 in the remaining simulations. Future work will explore a preferred value of $N_{\text{wait}}$ for larger FIFO_LENGTH parameters.

[29] A simulation investigating the effect of $N_{\text{blank}}$ has also been performed. In this study, two threshold levels are defined: a higher “detection threshold” ($\beta^2 = 90$) used for pulse detection as usual while $N_{\text{blank}}$ is varied. The output of the blanker is then examined to determine the number of samples remaining that exceed a lower “reference threshold” ($\beta^2 = 30$ using the same mean and variance computations as those for the detection threshold). This quantity is labeled $P_{\text{out}}$ in what follows. The total number of samples exceeding the reference threshold in the original data is also determined, and labeled $P_{\text{in}}$. The ratio $(P_{\text{in}} - P_{\text{out}})/P_{\text{in}}$ then provides information on the effectiveness of the higher threshold blanker with a given $N_{\text{blank}}$ at removing lower-level pulse contributions. Figure 4 illustrates these quantities for a single LISA capture. Note in some cases, samples exceeding the higher threshold are not blanked in this data set; this is because such “trigger” samples may be missed by the subsampled detector if they are less than 4 samples long.

[30] Figure 5 plots $(P_{\text{in}} - P_{\text{out}})/P_{\text{in}}$ as a percentage (solid curve) from the entire channel 7 data set, with $N_{\text{blank}} \geq 1536$ which clearly satisfies equation (5). The curve shows only a modest variation with $N_{\text{blank}}$, and an $N_{\text{blank}}$ value in the range 1536 to 2048 appears appropriate in this case. This can also be interpreted as indicating that pulsed interference longer than 76.8 \textmu sec

![Figure 4. Detection threshold (for detecting pulses) and reference threshold (for estimating number of pulses). $N_{\text{wait}} = 0$ and $N_{\text{blank}} = 1536$ are used in the APB output results shown. See color version of this figure in the HTML.](image2)
(i.e., 1536 samples) is not significant in this data. Of course, this $N_{\text{blank}}$ parameter will vary for RFI environments dominated by different types of sources, but the cross–U.S. flight considered here should be fairly representative of other data sets. Results for smaller $N_{\text{blank}}$ values may show more sensitivity in Figure 5, but are not explored here because of the limitation of equation (5). Data for smaller $N_{\text{blank}}$ values could be generated by varying the FIFO_LENGTH and $N_{\text{wait}}$ parameters, and will be explored in future work.

[31] The saturation of the solid curve at a maximum value of approximately 80% is caused by the presence of spurious internally generated interference in the LISA data set [Johnson and Ellingson, 2003]. The dashed curve is computed by redefining $P_{\text{in}}$ and $P_{\text{out}}$ to include only consecutive sets of 2 or more samples that exceed the reference threshold. In this case, a removal of approximately 98% of the “low-level” pulse components is achieved.

5.2. Output $\chi^2$ Test

[32] Now that means for choosing the APB parameters have been established, it is of interest to quantify the quality of the output data. Because thermal noise power should approach a Gaussian distribution when integrated sufficiently, the $\chi^2$ test against a Gaussian distribution [Press et al., 1992] can be utilized to evaluate if the output of the blanker satisfies this expectation. However, the entire data set for a specific channel cannot be applied in this test, as the mean noise power in a channel will vary as different locations are observed. The test was instead performed using sets of consecutive 5 16K captures, all of which were measured within 5 seconds. The $\chi^2$ statistic using 4 degrees of freedom was computed using data power integrated over 128 samples in order to approach the Gaussian distribution.

[33] Original data from one set of LISA channel 7 data is illustrated in the top plot of Figure 6 before and after the APB algorithm is applied with $\beta^2 = 40$ and $N_{\text{blank}} = 2048$. The presence of interference in the original data set results in a high $\chi^2$ value of 262.53, indicating that the data are not likely to be from a Gaussian distribution. The bottom plot illustrates the $\chi^2$ statistic of the nonblanked data after blanking with $\beta^2 = 40$ versus $N_{\text{blank}}$ and shows a greatly reduced value compared to the preblanking case. Critical values based on $\alpha = 1\%$ and 10% (the probability of incorrectly classifying a true Gaussian distribution as non-Gaussian) are also illustrated in Figure 6. Clearly for this example, the APB output data is much more Gaussian than the input data, particularly for $N_{\text{blank}}$ exceeding 1366. The poor performance for $N_{\text{blank}} = 1024$ is not surprising, since this case does not satisfy equation (5), allowing the possibility that some detected samples remain unblanked.

[34] Simulations from other LISA data subsets show similar results, with a few exceptions. In particular, LISA channel 6 (1310–1330 MHz) sometimes contains very strong interference from multiple aviation radars, and a large value of $\chi^2$ remains even after blanking with $N_{\text{blank}}$ up to 4096. The limitation of the fixed FIFO_LENGTH parameter is an issue here, and future work will examine if increasing this parameter can improve these problematic data sets. However it should certainly be expected that there are cases with exceptional RFI corruption for which the APB algorithm cannot retrieve the original noise power.

5.3. Effect of Blanking on Integrated Spectra

[35] The ideal pulse-blanking algorithm would remove only RFI information, without changing properties (par-
particularly the mean power level) of the remaining noise information. One questionable issue of the APB algorithm is the impact of forcing data to zero when pulses are detected. This introduces discontinuities into the signal which may lead to undesired effects on the final output, as well as calibration uncertainties. Note after the APB operation an FFT is performed in the interference-suppressing radiometer of section 2; clearly the impact of blanked samples on the FFT output should be investigated.

To examine these effects, APB outputs were processed through FFT and integration operations. Each 16K LISA capture after blanking was first separated into 32 512 sample “frames” (i.e., a 512 point FFT operation is used). Prior to the FFT, each frame can be categorized as either BLANK (contains no nonzero samples), NO BLANK (contains no blanked samples), or PARTIAL BLANK (some samples are blanked), as shown in Figure 7. The FFT is performed on each frame, the power computed in each FFT bin, and all results in each FFT bin are averaged.

It is clear that the only effect of the BLANK category is to decrease the noise power level of the final average. It is trivial to correct for this effect simply by counting the total number of frames and the number of BLANK frames. However, the effect of the PARTIAL BLANK frames, which contain discontinuities, is more complex. An FFT operation on such a frame clearly will produce a distorted spectrum and a reduced noise power level, with the degree of distortion and power reduction related to the number of blanked samples within the frame. Clearly, narrowband noise sources may experience some distortion in this process; however, noise sources with bandwidths larger than a few MHz, which are the subject of this work, are not considerably affected. An example PARTIAL BLANK spectrum is compared to the corresponding NO BLANK average spectrum in Figure 8. The reduction in power level is clearly visible, along with a moderate distortion in the overall shape of the spectrum. Note the large power levels observed near the center of this spectrum are due to an internal DC component of the measured power, not due to external interference.

Various means for coping with the PARTIAL BLANK issue can be conceived. A simple strategy (called method 1) is to eliminate such frames from the averaging operation; however this approach may also eliminate a large fraction of the incoming data in high-RFI environments. A second approach to retain these frames while correcting for the power level reduction is called “method 2: instantaneous scaling.” By Parseval’s theorem, the effect of blanking on total average power of the frame can be corrected simply by increasing the power of the computed spectrum by $N / N_{\text{rem}}$, where $N_{\text{rem}}$ is the number of nonblanked samples in the frame. This correction is applied to the power level of each frame before including the frame in the average computation. A final approach to is to included all (unscaled) frames in the spectral average operation, and to maintain a separate count of the total number of nonblanked time domain samples included in the average, labeled $N_{\text{tot}, \neq 0}$. Only the final average power is scaled by $N_{\text{tot}} / N_{\text{tot}, \neq 0}$, where $N_{\text{tot}}$ is the total number of time domain samples that make up the average operation. This approach is termed “method 3: slow scaling”.

The top plot of Figure 9 compares average spectra from a single Channel 6 LISA 16K capture before any blanking operations with the averages obtained from methods 1 and 2. APB parameters $\beta^2 = 90$ and $N_{\text{blank}} = 2048$ were used in the APB algorithm. Both blanking methods are seen to be highly effective in removing contributions from a pulsed interferer centered at approximately 1315 MHz in this example. Results from Methods 1 and 2 are also observed to be very similar, demonstrating that the instantaneous scaling approach is correcting for the reduced noise power level due to blanking effects. To highlight the differences between methods 1 and 2, the bottom plot of Figure 9 illustrates the difference (subtracted decibel values) between the method 2 (instantaneous scaling) and 1 (NO BLANK only) average spectra. Differences are generally within

![Figure 7. Definition of BLANK, PARTIAL BLANK, and NO BLANK frames. See color version of this figure in the HTML.](image-url)

![Figure 8. Frequency spectrum of an example PARTIAL BLANK frame, compared with the entire NO BLANK average for the corresponding capture. See color version of this figure in the HTML.](image-url)
1.5 dB in all cases, and appear noise-like, indicating that further averaging would likely make these differences less significant.

The difference between averaged spectra for methods 3 (slow scaled) and 1 (NO BLANK only) is also illustrated in the bottom plot of Figure 9; errors from method 3 are observed to be somewhat smaller than those from method 2 on average. Clearly method 3 is a simpler operation than that of method 2 (favorable for hardware implementation) since corrections are required at a much slower rate. In addition, method 3 should be preferable to method 2 because method 2 allows PARTIAL BLANK frames with a great deal of blanking to be weighted equally in terms of the power averaging computation. However, these frames also have the largest degree of spectral distortion, so reducing their weight should be advantageous. Figure 10 illustrates the mean error for methods 2 and 3 (compared to method 1) for LISA channels 7 through 13 when averaging results over the entire data set and the entire frequency spectrum. The mean error is of interest because it indicates the degree to which the average power level is not being corrected properly. Results clearly show the method 3 error (slow scaling) generally to be smaller than that of method 2. A hardware implementation of method 3 is currently in progress for the digital radiometer prototype.

5.4. Frequency Domain Blanking

Although the digital radiometer prototype does not implement RFI mitigation strategies in hardware after the FFT operation at present, it is of interest to simulate the expected performance of such approaches. The “channelization” of the FFT should allow an improved signal-to-noise ratio in detecting pulsed interference within a single FFT bin. A hardware blanking algorithm could conceivably operate on each FFT bin in real time by using a strategy identical to that of the APB processor. Such an approach would allow lower-level, rapidly pulsed RFI to be removed if missed by the original APB.

This algorithm was simulated using the LISA data of 145 captures in channel 6. After passing each capture through the time domain APB algorithm with $\beta^2 = 40$, and $N_{\text{blank}} = 4028$, each 16K capture was split into 32 512-point frames. An FFT operation was then applied to each frame, resulting in a total of $32 \times 145 = 4640$ temporal samples for each FFT bin. For each bin, a second APB algorithm with $\beta^2 = 90$, $N_{\text{blank}} = 4$, and $\text{FIFO\ LENGTH} - N_{\text{wait}} = 2$ was then applied to these 4640 samples, with the mean and variance computed by the averaging filter process used in the original APB algorithm. Figure 11 illustrates the average spectra before and after the frequency domain blanking operation, and shows a slight change in results near the center of the spectrum. The effect of the blanker is more obvious in the “max hold” spectra also illustrated in the plot; “max hold” refers to the maximum value of the 4640 temporal samples. Clearly a significant degree of RFI is included in this data set near the center frequency 1320 MHz; the frequency domain blanking operation reduces this interference so that corruption of the average spectrum is less significant.

An alternative approach, “$\chi^2$ blanking” can also be considered if blanking at a slower temporal rate is deemed acceptable. In this method, the $\chi^2$ test is performed on data from each bin. If the $\chi^2$ value exceeds a specified critical value, samples in the data set exceeding a power threshold are removed and $\chi^2$ then re-evaluated. This iteration is repeated until the distribution satisfies the $\chi^2$ test for a specified critical value. The requirement for a slower temporal rate here is due to the complexity and
iterative nature of this algorithm, which is not suited for integration in hardware. However, the algorithm could be applied to already integrated data as a postprocessing step in software.

6. Conclusion and Discussion

[44] Results of the study show the APB approach generally to be effective in reducing corruption from temporally localized RFI. The simulations performed on the LISA data set, while not completely general, arguably should be representative of a wide range of RFI environments. For these simulations, use of $\beta^2$ values ranging from 40 to 90 appears to effective, along with $N_{\text{blank}} > 1536$ ($\approx 76.8$ $\mu$sec blanking window). Use of “slow scaling” of the output power after blanking was found preferable for correcting errors in the estimate of mean power due to blanking, and averaged spectra after blanking showed only a modest distortion of the underlying noise power. Further studies will continue to explore the effect of other parameters in the APB algorithm (particularly FIFO_LENGTH), as well as studies with other data sets. In particular, an airborne version of the digital receiver prototype is currently in progress for RFI observations at C band [Johnson et al., 2004]. Results from flights of this system will provide a substantial set of RFI observations for further evaluating APB performance in varying RFI environments.

[45] Acknowledgment. This work was supported through the Instrument Incubator Program of NASA.

References


Ellingson, S. W., and J. T. Johnson (2003), Airborne RFI measurements over the mid-Atlantic coast using LISA, technical report, ElectroSci. Lab., Ohio State Univ., Columbus, Jan.


Johnson, J. T., and S. W. Ellingson (2003), Airborne RFI measurements using LISA during transit to and in the Wakasa Bay campaign, technical report, ElectroSci. Lab., Ohio State Univ., Columbus, July.


S. W. Ellingson, Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, 340 Whittemore Hall, Blacksburg, VA 24061, USA. (ellingson@vt.edu)

J. T. Johnson and N. Niamsuwan, ElectroScience Laboratory, Department of Electrical and Computer Engineering, Ohio State University, 1320 Kinnear Road, Columbus, OH 43212, USA. (johnson@ee.eng.ohio-state.edu; niamsuwan.1@osu.edu)