RTCA Special Committee 186, Working Group 3

ADS-B 1090ES MOPS

Meeting #19

RTCA, Washington DC
7 December 2005

New 1090 MHz Extended Squitter Error Detection and Correction

Dr. Jeffrey Gertz
MIT Lincoln Lab

SUMMARY

This Working Paper reports on work by Dr. Jeffrey Gertz to develop further improvements in the detection and error correction of 1090 MHz Extended Squitters.

Improved ADS-B Squitter Reception Algorithm

Dr. Jeffrey L. Gertz
October 24, 2005

Overview

As Mode S Extended Squitter messages are received, they are accompanied by interference from both ATCRBS and Mode S aircraft. Mode S reception techniques have been developed to obtain reliable performance even in very heavy interference environments. The interference environment is maximum when reception is omnidirectional, which is true for airborne receivers. The techniques developed for omnidirectional reception are documented in the Extended Squitter MOPS (DO-260A).

In the years since the DO-260A reception techniques were developed, avionics processing power has improved significantly. It is now possible to consider certain additional improvements in Mode S reception. As work on Mode S reception has gone forward, a variety of possible techniques have suggested themselves from time to time, which were not adopted originally because of the status of avionics processing power at that time. Certain individual reception patterns, particularly when a Mode S signal is overlapped by another Mode S signal, are candidates for further improvements in performance. Given the current processing capabilities, further enhancements in Mode S reception are now being developed.

This paper presents a first overview of a set of new procedures that have been found to significantly increase the detection probability of ADS-B replies in an interference environment. Performance comparisons are made to the DO-260A approach; approaches developed by the FAA Technical Center and industry have been found to provide comparable performance.

Current Lincoln Approach

The previous Lincoln Laboratory bit declaration algorithm employed a table lookup procedure. Each of the 10 samples per Mode S bit position were scored according to how their amplitude compared to the preamble amplitude:

0 = more than 6 db below the preamble
1 = between 3 db and 6 db below the preamble
2 = within 3 db of the preamble (either + or -)
3 = greater than 3 db above the preamble

The samples were then divided into odd and even sets (based on their sample index), and the 2 total scores used as indices into precomputed tables. (Note: the use of odd and even tables is a compromise -- having a single table using all 10 samples would have required

too much memory.) The possible results for each bit, after logic using the table values was applied, were:

H0 – high confidence ‘0’
L0 – low confidence ‘0’
H1 – high confidence ‘1’
L1 – low confidence ‘1’

Once the bit declaration was completed, the error syndrome of the reply was computed. If 0, the reply declaration was assumed to be correct, else a pair of error correction algorithms were applied that either corrected the reply or left it as uncorrectable. The algorithms employed, in order, were:

1. Conservative Mode S – attempts to find a set of low confidence bits in a 24 bit span that matches the syndrome
2. Brute Force – attempts to find a set of low confidence bits, from among all low confidence bits, that matches the syndrome

The “conservative Mode S” algorithm requires all confidence bits to be located within a 24-bit span, and as such has a negligible possibility of producing undetected errors (as Mode S employs a 24-bit cyclic code). The “brute force” algorithm is only attempted if the number of low confidence bits over the entire reply is no more than a parameter. This parameter is recommended to be set at 5 if undetected errors are a concern (the Mode S code Hamming distance is 6, so any lower values produces few undetected errors).

Proposed Improved Approach

The new improved approach begins by increasing the number of sample score values from 4 to 7 so as to provide greater resolution of the overlap interference patterns:

0 = more than 7.5 db below the preamble
1 = between 7.5 db and 4.5 db below the preamble
2 = between 4.5 db and 1.5 db below the preamble
3 = within 1.5 db of the preamble (either + or -)
4 = between 1.5 db and 4.5 db above the preamble
5 = between 4.5 db and 7.5 db above the preamble
6 = greater than 7.5 above the preamble

Next, instead of a single decoding table applied to all situations, a large set of specialized tables is maintained in addition to the general table. These specialized tables are applied whenever an overlap situation is detected by the declaration of a second preamble within the subject reply being decoded. The multiple decoding tables are indexed by the db level and the modular offset of the interfering reply preamble – for example, one table would apply to the case of an interferer 1.5-4.5 db higher whose bit positions begin 3 samples into the bit positions of the original reply. Thus there are 140 specialized tables – 7 db ranges x 10 bit samples per pulse position x 2 sample sets (odd

and even as described above). Each table contains 16,807 entries, for a total memory requirement of about 10 megabytes.

These procedures attempt to decode both replies in an overlap situation. Of course, this is possible only if the second preamble is actually declared, which is presently unlikely if the second reply is not higher power than the first. To increase the chances of preamble declaration, the new approach also includes a method to locate and declare preambles of overlapping replies when the preambles are obscured by the initial reply. The approach used is to find the reply's end pulse, which will likely be in the clear unless a triple overlap exists, and work back to identify the preamble – a sort of postamble detector. Should a preamble be possible at the indicated position, it is declared, with its amplitude given by that of the final pulse. Since reply detection would now only commence at the end of the pulse string, a delay is added to the output latency, although the delay will not exceed 250 microseconds.

Initial Results

To measure the improvement resulting from these algorithm enhancements, the following scenario was tested:

1. A subject ADS-B reply was constructed with a power level uniformly distributed from –90 dbm to –60 dbm
2. An interfering reply was added, also with a power level uniformly distributed from –90 dbm to –60 dbm
3. The interfering reply preamble time was uniformly distributed in the interval + or - 120 microseconds from that of the subject
4. An ATCRBS fruit was added to the mix, with the same power distribution, and located from –20 microseconds to +100 microseconds relative to the subject reply
5. Relative frequency and phasing of all replies was uncorrelated, and all reply codes were randomly chosen

A million simulation trials were run, and the results segregated according to whether the subject reply was the first or second of the overlapping pair. This dichotomy lets us determine the affect of the postamble algorithm on bringing the detection performance of the second reply near that of the first reply.

Figures 1 and 2 present the results observed from the simulation, when the subject reply was the first and second of the overlapped pair respectively. Two conclusions are evident. First, the proposed new method offers significant improvement in ADS-B squitter reception, particularly for low power replies. Second, the second reply of the overlap pair no longer suffers significant degradation in reception relative to the first.

Further Reception Performance

Additional improvement in decoding performance is possible by becoming more aggressive in the error correction attempts. Two enhancements are now under consideration, both to determine their improvement potential and to evaluate whether the inevitable increased undetected error rates are acceptable.

The first, and most obvious, approach is to increase the brute force parameter above 5. With increased computer speed, higher values may now be computationally feasible. A value of 16 is currently being tested.

The second enhancement is a new algorithm dubbed “fixbits”. This procedure attempts to precompute a set of bits to correct for each possible error syndrome. This attempt is considered successful if a unique bit set is found that has fewer members than any other set that also satisfies the syndrome. The use of the procedure requires the following criteria to be satisfied:

1. The reply has fewer than a parameter number of bad bits, defined as those that have no detection information.
2. The bits in the correction set contain no more than a parameter number of high confidence bits; the parameter is typically set to 0 or 1.

Figure 3 shows the improvement potential of these aggressive error correction algorithms using a parameter value of 1. The accompanying undetected error rate is still under investigation. Of course, a density and power model of Mode S replies and squitters, as well as ATCRBS replies, is required in order to compute the error rate; one is now being developed.

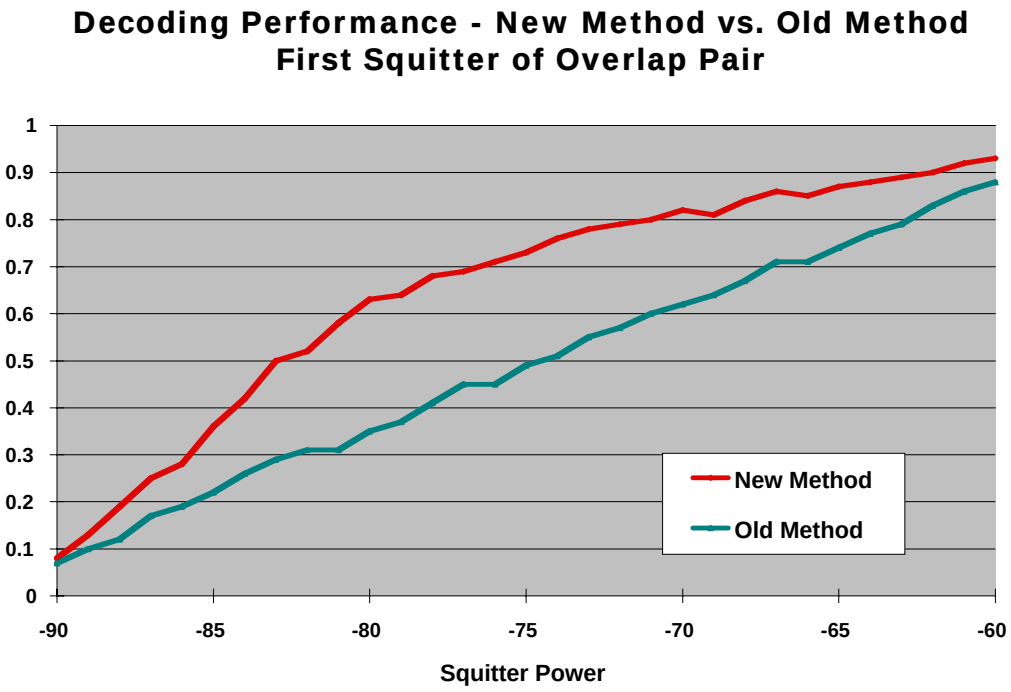
Decoding Performance - New Method vs. Old Method
First Squitter of Overlap Pair
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
-90
-85
-80
-75
-70
-65
-60
Squitter Power
New Method
Old Method

Figure 1.

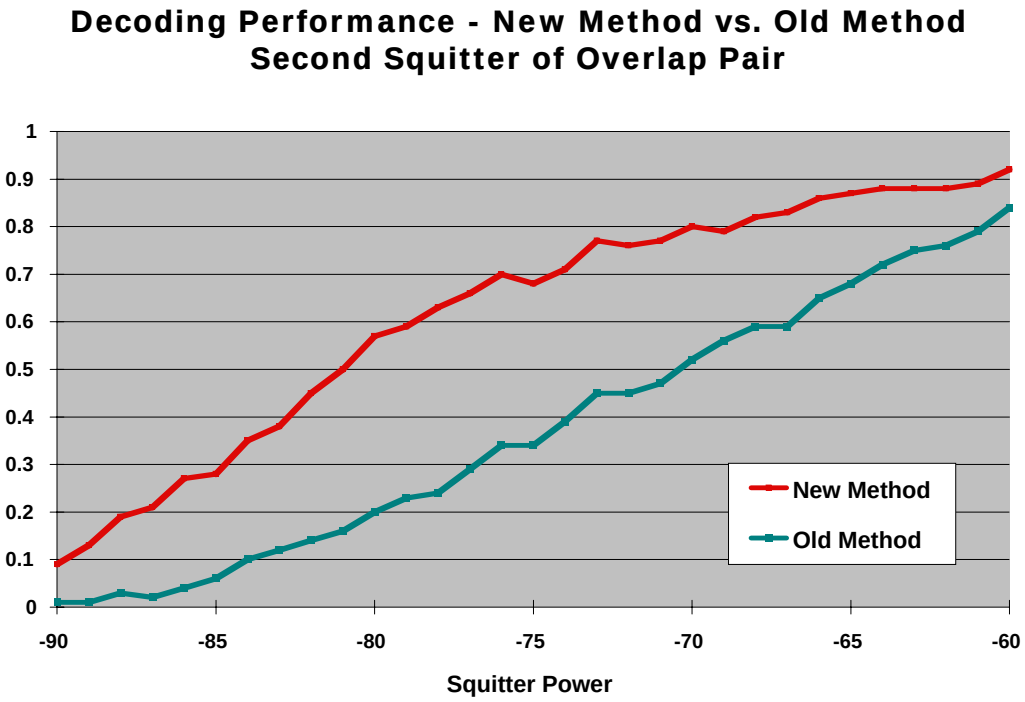
Decoding Performance - New Method vs. Old Method
Second Squitter of Overlap Pair
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
-90
-85
-80
-75
-70
-65
-60
Squitter Power
New Method
Old Method

Figure 2.

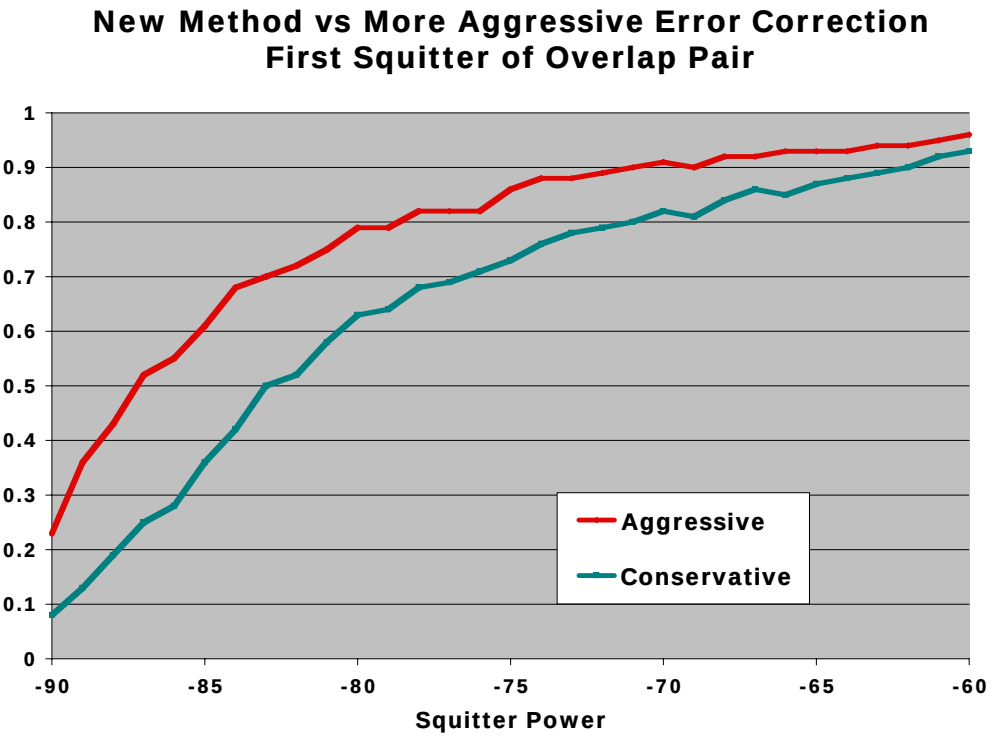
New Method vs More Aggressive Error Correction
First Squitter of Overlap Pair
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
-90
-85
-80
-75
-70
-65
-60
Squitter Power
Aggressive
Conservative

Figure 3.