Interactive comment on "Eddy Covariance flux errors due to random and systematic timing errors during data acquisition" *by* Gerardo Fratini et al.

Anonymous Referee #1

Received and published: 10 May 2018

This manuscript explores the effects of data stream synchronization on fluxes derived from all-digital data acquisition systems. While the methods and conclusions seem sound, there is little practical information about how this work applies to real-world systems such as TK3, EasyFlux, EdiSol, HuskerFlux, or SmartFlux. This work would be of greater value if the authors could review some of these systems and comment on whether or not the issues they explore are present or absent in any of these packages.

Never the less, I believe that this work does have value to the eddy covariance community and should be reported with several modifications.

1.) throughout the manuscript, "prospect" is used when "prospective" is appropriate 2.) on pg. 1, line 27, please define the term "zero-hold" 3.) on pg. 5, line 11, change "AT clocks" to "AT cut crystals".... also throughout the manuscript, please don't confuse

the term "clocks" (a system) with "crystals" (a component of a system). 4.) on pg. 6, line10, what do the authors mean by the term "vector"? A vector is a quantity that has magnitude and direction. How does this apply to time? 5.) on pg. 8, line 16, do the authors mean to use the term "filter" in this context? Does this imply that a mathematical operation was applied to the data in figure 5? 6.) on pg. 9, line 5, see #3 above

There may be other instances of undefined or confusing terms that I've missed. I'd encourage the authors to carefully review the manuscript for this.

Other more general comments follow:

The authors imply that all-digital data acquisition is a very recent development. This is not true. I've been aware of all-digital solutions for at least 15 years. One in particular (HuskerFlux from U. Nebraska or maybe Lawrence Berkeley Lab, I can't quite remember now) seems to have addressed a number of the issues identified here such as re-synchronization of data streams. The authors also imply that many of the synchronization problems outlined in the manuscript are absent from analog data acquisition systems, but this is not exactly true. Because of the "sample and hold" nature of A/D systems, many of these issues while present are masked. The authors also suggest that Ethernet connectivity is also relatively new, but again, this has been available for a long time, especially in Campbell Scientific data loggers (via the NL-100 module). One issue that the authors identify in serial data communications are the FIFO buffers used by many operating systems to ingest RS232 data. While these do exist and would create problems, well designed programs often get around this by lowering the size of these buffers and/or running independent program "threads" that handle individual character-received interrupts to pass the data along in near-real time. The authors also state that STE timing issues are not detectable, but I must disagree. When testing several data acquisition packages, I found that the HuskerFlux package recorded the individual buffer size differences after a user chosen interval. This difference can be used to calculate the magnitude of the STE over that interval. This should be relatively

easy for any new software to do. Finally, in developing a method to check any particular system for timing errors, the authors suggest using a signal generator to inject a single waveform into the A/D input of both instruments while having one instrument also send the same signal from it's D/A outputs to the second instrument. While this will work in principle, it must be cautioned that this is only strictly true if the D/A task and the A/D task are both synchronized with the measurement task and the serial output task in both instrument firmwares. Will this always be the case, or is this only true in some instruments such as the LiCor and Gill units tested by the authors? If these tasks are only loosely synched or are running asynchronously, then some issues could be masked by the internal asynchronicity.

I would encourage the authors to consider these issues and perhaps explore some of the available all-digital solutions to perhaps assist users in choosing a system for their purposes.

C3