Re: how to find the best ADC step size?

Source: http://sci.tech−archive.net/Archive/sci.math/2004−12/9922.html

---

From: Johan Carlson (*Johan.NOSPAM.Carlson_at_csee.ltu.se*)
Date: 12/22/04

Date: Wed, 22 Dec 2004 23:13:54 +0100

Randy Yates wrote:

> "Clay S. Turner" <Physics@Bellsouth.net> writes:
>
>
>>"Randy Yates" <randy.yates@sonyericsson.com> wrote in message
>>news:xxpllbqjpjf.fsf@usrts005.corpusers.net...
>>
>>>"Clay S. Turner" <Physics@Bellsouth.net> writes:
>>>
>>>>[...]
>>>>The idea is to try to make each symbol contribute equally to the overall
>>>>process. Imagine looking at your data after a huge number of symbols was
>>>>received. The idea is to make the info provided by each type of symbol
>>>>contribute equally.
>>>
>>>But this Huffman coding won't do that. Choosing a representation for a
>>>symbol doesn't change the probability of the symbol occurring. It
>>>does, however, minize the average symbol rate − I certainly see
>>>that. Perhaps I'm being blind?
>>>−−
>>
>>Hello Randy,
>>
>>The Huffman problem and the quantization problem are related in they both
>>implement methods to effectively flatten out variations. True they are
>>different in that the Huffman problem is trying to minimize the average
>>number of bits per sample and the quantization problem is trying to maximize
>>the information per sample. The differences arise from the constraints and
>>what you are starting with. In one case we have a fixed amount of
>>information, so how few total bits can we fit all of the info in. The
>>theoretical answer is the entropy, but the practical answer (comes from
>>requiring whole numbers of bits in a symbol is the Huffman entropy.) In the
>>other case we have a fixed symbol size, so how can we get the most info per
>>symbol. The connection is the entropy and its maximization requires a flat
>>distribution.
>>
>>In Huffman coding we sort the symbol probabilities into descending order and

>>then we make multiple passes throught the list each time combining the two
>>lowest probilites together. So we are essentially trying to make each path's
>>probability be the same or at leat similar. In the quantization we are just
>>diving up the total probability into N equal regions.
>>
>>You are not being blind, you are just asking how two seemingly different
>>things are related. And the relation is through a flattening out of the
>>probabilty function.
>>
>>I hope this helps to clear some things up.
>>
>>Clay
>
>
> Intriguing stuff! I'm not sure I follow the sorting description, but
> that's OK. I need to go back and read fully and with undrstanding
> the seminal Shannon papers.

Just a thought... don't know if it's at all relevant to the thread, but
here we go anyway:

I you do a Huffman coding, common symbols are given a longer code word.
This will minimize the average data rate. If you adjust quantization
similarly, I get the feeling this will also minimize the average
quantization error.

/Johan