Hydrol. Earth Syst. Sci. Discuss., 10, C1680–C1692, 2013
www.hydrol-earth-syst-sci-discuss.net/10/C1680/2013/

Hydrology and Earth System Sciences
Discussions

Interactive comment on "Data compression to define information content of hydrological time series" *by* S. V. Weijs et al.

S. V. Weijs et al.

steven.weijs@epfl.ch

Received and published: 13 May 2013

We thank the reviewer for his/her thoughtful comments, questions and suggestions, which will be very helpful in improving the paper. Below we respond to the comments (*repeated in italic font)* point by point.

General remarks: To derive the information content of hydrologic series, the article explores the potential of a number of data compression methods which are broadly adopted in computer sciences, signal and image processing as well as telecommunications to reduce data storage and transmission capacities. In computer science and signal processing the linkage with information theory lies in the fact that data compression (and decompression) needs to be achieved with a minimum loss of information.

In hydrology, information theory has been so far adopted using entropy concept in hydrological frequency analysis or to size hydrological networks (decision problems) or to adjust rainfall-runoff models using specific cost functions and more recently as a way to define the information content of data in regionalization or sampling problems. Therefore, the linkage between data compression technology and hydrological regionalization which is the main focus of the paper is really a very interesting and vast domain of investigation in hydrology.

We are pleased that the reviewer shares our idea that the data compression view opens up a vast domain of ideas to explore. We will make an effort to revise the paper so it becomes more accessible to hydrologists without previous experience in information theory.

However, it is not very frequent to find entropy expressed in bits neither in hydrological frequency analysis nor in network optimization problems. So, I found that the style of paper and its writing do not help readers from the hydrological and climatological fields to understand the potential of the connexion with computer sciences and signal processing disciplines. Therefore, my general recommendation would be to "smooth" the text in order to make it easier for readers from the hydrology side (as supposed are HESS readers) and to give more details about the hydrological context.

We also found that entropy in hydrological literature is often measured in nats, not bits. However, bits have the advantage over nats that they have a clear interpretation, both in data compression and in terms of uncertainty (e.g. number of optimal yes-no questions to obtain the answer). While it is just a change of unit, we think in this case the bit is more suited than the nat, since file sizes are also measured in bits. We detailed this in an expanded short introduction of information theory aimed at hydrologists, which will also mention that 1 nat = 1.443 bits. Throughout the paper, we will add some more clarifications of terminology and aid interpretations for hydrologists.

In order to help interpret the information content obtained it seems important to give an

idea about hydrological data at least for the Leaf river basin (some basin physiography characteristics, sample statistics and main hydrological signatures). For basins from MOPEX project, we would need to give an idea about basin climate classification, area, average annual rainfall and runoff etc. . . which represent the hydrological context of the study. Otherwise, this article would be more suitable in an informatics journal, addressed to capture informatics scientists about the potential of their methods in the hydrological field.

We will add the ranges and some statistics on the Leaf river and Mopex data set to sections 3.4 and 3.5. We also will discuss more elaborately the links between hydrological characteristics and compression results. We believe that especially for the publication in a Hydrology journal, we should focus on explaining IT and AIT, which are new for the readers. The hydrological context of MOPEX and Leaf River data sets have been presented before in the hydrological literature, so we think it would be best to restrict the discussion of that context to what is necessary to interpret the results. We will also make clearer in the paper that we will focus on inherent limitations in assessing information content, rather than directly interpreting the results in a hydrological context. To connect the paper better to hydrological practice, we will discuss some possible uses of the results on information content, when the limitations have been properly dealt with by asking the right question. We will also present some climate/hydrological information (e.g. the Köppen climate classes) that partly explains compression results as a table for all basins in the supplementary material.

Specific remarks (P 4) The introduction of 2.1 is difficult (lines 8 to 14). A definition of the code function seems to be necessary before introducing code lengths. Also, the definition of the prefix code would be helpful at the beginning of the paragraph.

We will introduce these concepts more gradually in the revised manuscript, with more reference to the figure of the binary tree. We will start 2.1 with a text along these lines: For the compression perspective, data can be regarded as a file stored on a computer, i.e. as a sequence of symbols that represent events or values that correspond to quan-

tities in a real or modeled world. Data compression seeks more efficient descriptions for data stored in a specific format, so they can be stored or transmitted more efficiently. Generally, this is done by exploiting patterns in data. One of the patterns that is often used for compression is the fact that not all symbols or events are equally likely to occur. In this paper, we focus on lossless compression as opposed to lossy compression. This means that we look exclusively at algorithms that can reproduce the original data exactly. Lossy compression achieves further compression by approximating the data instead of generating them exactly.

Data compression seeks to represent the most likely events (most frequent characters in a file) with the shortest codes, yielding the shortest total code length. There is a close connection between code lengths and probabilities, as explained in the following.

The binary tree of Fig. 1 should be well introduced (the size, the height, the nodes, the roots, the leaves, branches . . .) in linkage with data structure and not only on the basis probability. The term event should be defined in relation with the tree representation as well as the length of the code (tree height, depth (distance to root node), leaves, nodes). It is not relevant to put such details in the appendix.

We will add: The binary tree illustrates the connection between code lengths and probabilities for prefix free codes. When the binary codes for data are concatenated in one file without spaces, and they have different lengths, they can only be unambiguously deciphered when no code in the dictionary forms the beginning of another code of the dictionary. This can be visualized as a binary tree, where the prefix-free codes must be at the leaves, since any codeword on an intermediate node is the prefix of all codes on the downstream leaf nodes. The depth of each branch represents the length of the corresponding code-word. The corresponding optimal probabilities of the events the words encode are $2^{-}L$. Another way to interpret these optimal probabilities is the idea that every branching represents one yes-no questions whose answer is encoded in one bit of the code word. These questions are optimal if they represent 50

Line 20 “see fig 1 code A” is not enough explicit of what you mean; you should explain this example in more details.

We will add: The four 2-bit binary numbers of code "A" are prefix free, since none of the codes is the first part of one of the other code. A sequence of these code words, without "spaces" in between can therefore unambiguously be decoded if the "dictionary" is known.

Huffman coding needs to be better introduced and documented as well as Range coding (P16 line 4).

We will give references for Huffman and Range coding at the point where they are first introduced at page 6. We will explain the general idea that both try to achieve the entropy bound, but do not explain the detailed workings of the algorithm, since these are not relevant for the rest of the paper and can be found in cited references.

The Kraft inequality (Eq. 1) and Eq. 3 (Kraft- MacMillan theorem) may be presented in a more comprehensive way. In particular you should give the definition of the Kullback-Leibler divergence in simple words (“a measure of the information lost when q is used to approximate p”) in this part of the text.

In the revised manuscript we will introduce KL divergence by giving its definition both as an equation and in words. We also added the summarizing remark at the end of 2.1 that Entropy gives a bound for the minimum file size achievable by making use of the distribution.

P5 line 2 the terminology of “bit per sample” might be specified here.

We noticed that the more common meaning of bps, is ”bits per symbol” so we changed the text and also explain it as the average number of bits per encoded value, i.e. per time step of the time series.

I think that the Appendix is not enough for the reader who is not used with this terminology or who is interested to go further in its application for his own purposes.

We tried use less specialized terminology where possible, but the appendix is aimed at readers who have familiarized themselves with the content of the referenced paper in Monthly Weather Review, since the appendix is about a data compression interpretation and extension of the decomposition presented there. Readers interested in forecast verification could refer to the MWR paper and other readers can skip the appendix without losing the main messages of the current paper. We added to the text: ”In the remainder of this appendix, we assume the terminology of that paper known to the reader. This appendix can be skipped by readers not interested in the connection to forecast verification.”

P5 in eq(2) you might specify the base of the logarithm (base 2 because you use the bits units?)

Indeed we use log2, we now specified this in the paper.

P5 line 10 the equation you are referring to is not specified (Eq. 3).

Corrected

P11 EQ. 5 minx , maxx and xinteger are not specified

Corrected: Minimum of x; maximum of x; x mapped to an integer in range 0-255.

P11 line 15 “The compression algorithms will be mainly used to explore the difference in information content between different signals “. How did you explore this idea in the results analysis?

This remark is to indicate that we do not need to interpret file sizes as absolute measures of information content. Even though we do not include the algorithm size, the results are not relative to hydrological prior knowledge, so fairly objective when used for comparison between hydrological time series. Still, the interpretation as relative measures of information content is not straightforward, as elaborated in the discussion section; hence we do not dwell too much on interpretation of the results. In the revised paper, we will add some discussion on what can be done with these measures of in-

formation content, and give some possible interpretations in the discussion section, so the purpose of the approach becomes clearer. We will also adapt the text on page 11, so it does not suggest to give a

P13 line 20 the generation procedure should be described shortly (sinus etc. . .?)

We noticed that we forgot the reference to figure 1. We will also include the Matlab code to generate the signals in supplementary material.

Line 22 14610 potential evapotranspiration (it is not potential evaporation)

Corrected

P14 l 12-14: the text is not clear

Replaced by : In order to losslessly reproduce Q, we could store P, a rainfall runoff model and the error time series needed to correct the modeled Q to the original measured time series. This way of storing Q and P leads to compression if the model is sufficiently parsimonious and the errors have a small range and spread (entropy), enabling compact storage. Since the model also takes some space, it is a requirement for compression that the error series are more compressible than the original time series of Q. In this experiment we test whether that is the case for the HYMOD model applied to Leaf River.

P15 line 7 the byte definition might be recalled otherwise the understanding of number 256 in "256 unit values" would not be direct.

We changed it to : one byte (8 bits), allowing for $2^8 = 256$ different values.

P15 line 10 "by value" has to be removed

" Value by value" was meant as "one value at a time" or one codeword per value. We reformulated to make it clearer.

P16 line 10 I could not understand your findings. Low entropy results in high pre-

dictability. In your case, are streamflow series more predictable or less predictable than precipitation series?; In Table 2 H/log N for LEAFQ (=42.1)>H/log N for LEAFP (=31) indicating that streamflow series are more unpredictable than rainfall series.

Indeed low entropy is generally an indication of high predictability. The entropy is not the whole story, since temporal dependencies can increase predictability and therefore compression, without being visible in the entropy of the signal. So P has lower entropy than Q and is therefore more predictable when knowing only the marginal distribution. However, Q has the lower conditional entropy given the value at the time step before, and is therefore more predictable than P when knowing the previous value each time the new value is coded/decoded. The compression algorithms can make use of this by for example first taking the lag-1 differences and then use standard compression techniques on that series, which can be transformed back to the original using a cumulative sum after decompression (We applied this in Weijs et al. 2013). As discussed under 3.1: Quantization, we must note that the results are (and should be) very dependent on the quantization, and direct comparisons of information content and predictability should be interpreted in the context of those quantizations (or "questions asked").

It is also the case in Fig. 3 for Mopex watersheds. How would you explain these results while the autocorrelation in rainfall series (=0.15) is far less than in runoff series (=0.89)

For the Mopex watersheds, entropy of Q is generally even higher than for Leaf River, because we first log-transformed the discharge. This results in values more evenly distributed over the 256 possibilities (or histogram bins), hence higher entropy. Here again, the entropy does not take into account temporal structure. So the high autocorrelation will result in better predictability and compression when the previous values are used to predict/compress the current. This is actually also one of the reasons why the data compression framework is more general than just looking at entropies: it naturally accounts for temporal dependencies. Results will still depend on the precise data compression algorithm used, but looking for better and better compressors will tighten the upper bound for estimation of information content. We adjust the text to reflect above

explanations.

(Table 3)? What kind of daily time series did you adopt for precipitations? Did you use spatial average daily rainfall for a given watershed or a single raingage in the outlet of the watershed or inside the watershed?

The precipitation used was area-averaged precipitation derived from several rain gauges. We will add this to the paper, with reference to the MOPEX data set documentation.

Did you control the fact that precipitation and discharge data are well compatible (by calculating runoff coefficients for example)?

The data set has been extensively used in previous studies (Vrugt et al, 2003; Schaefli and Gupta, 2009), and showed reasonable performance with conceptual mass-conserving hydrological models. We did not do any further checks, since for the present study the data set is just used for illustration of the ideas, and data quality is not critical for that purpose.

For the lossless compression context as stated in p9 line 12., the same quantity of information as the original is carried out using fewer bits; this leads to more information per bit which is equivalent to more entropy.

In fact, the algorithmic approach to information content, such as used in AIT, can be seen as an alternative framework for information content, which also measures information in bits, like Shannon's entropy does. In AIT, the Kolmogorov complexity of x, K(x), is analogous to the entropy of X, H(X). In fact, the analogy goes very far, as demonstrated in the Chaitin (1975) paper. The difference is that the AIT framework has more flexibility to describe patterns and takes into account the complexity of those patterns. Another important difference is that Shannon entropy is defined for a random variable, of which a particular sequence is thought to be one realization, while Kolmogorov complexity is defined for an individual sequence, without referring to an

underlying generating distribution. So if we can find a description shorter (measured in bits) than the entropy (also measured in bits), we should see this as a sharpened upper bound for the information content, which is measured by the number of bits description length.

Does Fig. 3 represent the information per bit ? On the other hand, the term better compressible should be explained (comparison of bit of information per bit of message or compression size normalized by entropy?).

We assume you refer to Fig. 4 here? Indeed the term better compressible in the caption is confusing, since actually P is better compressible than Q when looking at file sizes not normalized by entropy. Figure 4 gives a dimensionless number (bits/bit) that is a measure for the amount of temporal structure. This number should in theory range from 0 to 1 for completely dependent to completely independent values respectively. The values larger than 1 are due to the fact that compression always has some overhead compared to the entropy, and especially in case some events have a probability larger than 0.5, e.g. 0's in precipitation series. An alternative way to calculate this temporal dependence indicator while compensating for this overhead is to normalize by the compressed size of the permuted timeseries (equal to entropy+overhead) instead of by the entropy.

The better compressibility of streamflow data should be interpreted in this part of the text (such as to be linked to watershed size and basin geological features) (Fig. 4a)

The increased temporal dependence follows from the low pass filter behavior seen in most watersheds, which dampens out fast fluctuations present in rainfall. This results in a loss of information. On the other hand, information is added by other dynamics like evaporation. Furthermore, the dependence on choices in quantization and differences in overhead make it difficult to directly compare and interpret temporal redundancy in P and Q. The issue of overhead can be remedied by normalizing compressed size with the compressed size of the permuted time series instead of with the entropy (see

response previous to comment).

P17 line 11 what is the purpose of the study of errors compression? Rainfall-runoff discharge errors are generally autocorrelated. What does it indicate relatively to model structure or performances?

The purpose of this study is to test the idea that the errors should be more compressible than the original series. The compressed size of the errors says something about the remaining uncertainty about Q when having the model and P. An intuitive way to reach this interpretation is the following: 1. The file size of errors estimates the information content of the errors 2. If you have the model predictions, you need the errors to reconstruct the original signal of Q. 3. So the information content of the errors can be seen as the missing information, i.e. uncertainty about Q, when knowing Qmod. Which can then be interpreted as the predictive uncertainty. Caveat: This assumes an additive error model, in which H(Q|Qmod) = H(Q-Qmod). The real predictive uncertainty = H(Q|Qmod), but the calculation of that quantity suffers from the complexities mentioned on P17(2045):5-9

Actually, a more refined interpretation of the compressed errors file size is related to potential model performance. For example, a constant bias would result in low entropy, high compressibility, hence low remaining uncertainty, but a model is only considered good if its predictions are unbiased. Also, if errors are highly compressible due to autocorrelation, that is an indication of potential for further model improvement. This can be achieved adding a fitted AR error model. Without such an error model, the true predictive uncertainty represented by predictive errors will be better described by entropy (compressed size without using temporal dependence structure), since thet potential for improvement is not utilized. We will mention mention this in the Aleatoric-Epistemic uncertainty discussion, linking it to a recent paper (Gong et al, 2013).

Lower entropy of the errors means lower unpredictability of errors. Here you are right to mention that interpretation of entropy in terms of data compression is not simple;

Indeed this interpretation is correct. Low entropy can also be interpreted as lower uncertainty/lower missing information about the errors; see explanation at the previous point. If for the same error entropy, the compressed size of the errors becomes smaller, i.e. the errors are temporally dependent, the missing information about Q becomes less, since part of it can be filled in based on the previous error. This can be exploited by using e.g. an AR1 error model next to the hydrological model. This would reduce predictive uncertainty for Q in this case. We agree that interpretation of information theory applied to real problems is not simple and many subtleties should be considered. The strength of the compression framework is that many of the subtleties relating to model complexity and dependencies are naturally accounted for and map intuitively to file size, whereas when working with entropies/conditional entropies, these things can easily be overlooked (e.g. the issue with the 256^2 bin histogram estimated from 14610 values).

P17 line 15 too complex

We will "uncompress" the sentence a bit to make it clearer: The conditional entropy H(Q| Qmod) gives a theoretical limit for compression of Q when Qmod is known, when no use is made of temporal dependencies in Q that are not modeled in Qmod. It must be noted, however, that this compression assumes the joint distribution of Q and Qmod to be known. If not known a priori, as is the case in practice, the joint distribution must be stored in the compressed file and will add to file size. If enough data is available/compressed, the extra amount of storage will be negligible, but this is not the case here, hence the conditional entropy underestimates the real information content of errors and therefore model performance would be overestimated if complexity is not taken into account. In the compression framework, this is done automatically because the compressed file contains all information necessary to decode, including the joint distribution if used.

References

Chaitin, G. J.: A theory of program size formally identical to information theory, Journal of the ACM (JACM), 22, 329–340, 1975.

Vrugt, J. A., Gupta, H. V., Bouten, W., and Sorooshian, S.: A Shuffled Complex Evolution Metropolis algorithm for optimization and uncertainty assessment of hydrologic model parameters, Water Resources Research, 39, 1201, 10.1029/2002WR001642, 2003.

Schaefli, B. and Gupta, H. V.: Do Nash values have value?, Hydrological Processes, 21, 2075–2080, 2007.

Weijs, S. V., Van de Giesen, N., and Parlange, M. B.: HydroZIP: how hydrological knowledge can be used to improve compression of hydrological data, Entropy, 2013.

Gong, W., Gupta, H.V., Yang, D., Sricharan, K., and Hero, A. O.: Estimating epistemic & aleatory uncertainties during hydrologic modeling: An information theoretic approach, Water Resources Research, in press, 2013.

Interactive comment on Hydrol. Earth Syst. Sci. Discuss., 10, 2029, 2013.