Hydrol. Earth Syst. Sci. Discuss., 9, C3955–C3962, 2012
www.hydrol-earth-syst-sci-discuss.net/9/C3955/2012/

Hydrology and Earth System Sciences Discussions

Interactive comment on "A data based mechanistic real-time flood forecasting module for NFFS FEWS" *by* D. Leedal et al.

D. Leedal et al.

d.t.leedal@lancaster.ac.uk

Received and published: 24 August 2012

Response to Dr Romanowicz

Many thanks for the time and effort reviewing this article. Also thank you for the supportive and useful comments. Below we hope to address the issues you have quite correctly identified.

Improved description of how the mathematical interpretation is applied

In the proposed revised manuscript the mathematical detail of the transfer function modelling, the conversion to state space format, and the data assimilation algorithm based on the Kalman Filter have been moved into an appendix to allow a clearer reading flow. The appendix includes a better description of the way the various equations are implemented in the final computational scheme including strengths and weaknesses. A short section is included to read:

"For transfer functions with a numerator and denominator order higher than one and two respectively, the mapping to an equivalent network of 1st order components is ambiguous. For example the second order TFs identified in the Eden case study could be mapped as a feedback structure; however, such a configuration would be considered non-mechanistic unless there is strong evidence to the contrary."

Re-arranged input non-linearity and state space description ordering

Within the proposed new appendix section the description of the input nonlinearity function is now before the description of the state space formulation as suggested.

More detail about the optimization process

As pointed out the estimation of the input nonlinearity function is a two stage optimization process whereby at each iteration of the adjustments to the input nonlinearity parameter set, a new optimal TF is estimated using an embedded call to the RIV function. In this way the input nonlinearity parameters and the transfer function parameters evolve together. The significance, and a brief description of this process, is included in the proposed new appendix section.

There are several parameter groups that require optimization:

- Transfer function
- Adaptive gain function
- Data assimilation hyperparameters
- heteroskedastic variance function

HESSD

9, C3955–C3962, 2012

Our approach was fairly straightforward but rather longwinded to describe, we carried out the following steps for each of the 6 model nodes: We first identified and estimated the TF structure and parameter values for a purely linear-in-the-parameters model using the **RIVID** and **RIV** algorithms in order to get a first-approximation at the TF model. We then used a 1st order approximation of this model within a State Dependent Parameter **SDP** algorithm (again part of the **Captain** Matlab toolbox) to *identify* an approximate, non-parametric, input nonlinearity shape. We then used a Piecewise Cubic Hermite Interpolating Polynomial (PCHIP) spline to parameterize this initial input nonlinearity shape. The PCHIP spline used 10 knots spaced logarithmically along the x-axis up to a 5m depth. Keeping the x-coordinate of the PCHIP knot locations fixed, we then optimized the y-coordinates using Matlab's **LSQNONLIN** implementation of the Interior Trust Region optimization algorithm. As mentioned above, this step also included an embedded **RIV** algorithm to ensure the residual series was always generated from the most optimal TF model given the present iteration of the input non-linearity function. The steps described above provide the input nonlinearity parameterization together with the *a* and *b* parameters of the TF. We then took the pragmatic decision that these parameters are at least close to optimal and no further optimization is performed on them. An alternative approach (also tried) would have been to incorporate these parameters into the data assimilation algorithm together with the Kalman Filter hyperparameters (state noise variance ratios, and heteroskedastic variance function parameters), and optimize all together in a large-scale optimization process. However, in testing, we found this large-scale optimization either: (1) would not converge on a solution, or (2) produced parameter estimates with a very large variance suggesting the cost surface is a big challenge for the optimization algorithm. In contrast retaining the TF and input nonlinearity parameters from the first stage and limiting the second optimization to the second order heteroskedastic variance term and the diagonal terms of the NVR matrix, resulted in much more consistent and robust optimization results that were less sensitive to initial parameter estimates. Following a scaling by the proportionality constant as shown in Equations 6 and 7, the resulting uncertainty range was tested and found to bracket a good approximation to the expected percentage of observations. However, please see below for issues relating to the uncertainty range for short lead times. We believe we have developed an optimization approach that balances heuristic decisions with state of the art algorithms to arrive at a workable, robust and pragmatic implementation. However, and not limited to the example presented in this paper, we feel applied optimization for complex multi-component non-linear models is an active area of research and we would welcome appropriate developments in this field.

I appreciate the multi-stage process described above is rather hard to communicate, especially in moving from a purely mathematical to an implementation-based description. The contact author would be happy of course to exchange further communications and forward the Matlab and R scripts used to produce the paper if at all helpful.

Some of the above description (in a more concise form) has been included in the new appendix section of the proposed revised paper.

Short lead time uncertainty estimates

The optimisation of the **Q** matrix and heteroskedastic variance parameters is carried out for a specific f-step ahead forecast (in the Eden case study this is the maximum lead time available for each node). This means that the estimates for the uncertainty range at shorter lead times will be conservative. However, both you and Reviewer 2 have drawn our attention to the rather larger uncertainty range shown in the figures. After an investigation we believe we have found an error in the Matlab implementation used to produce the figures for the paper, where-by the scaling factor was not applied to the 2nd order term in the heteroskedastic variance function (this problem is not present in the FEWS R coding). Secondly we have performed the optimization for the f-step lead time shown in each figure to provide a more realistic visualization of the estimated uncertainty range at that specific lead time. The new figures are attached.

Discussion of specific requirements of an online algorithm

HESSD

9, C3955–C3962, 2012

Interactive Comment

The reviewer correctly points out that the paper does not provide enough detail of the online nature of the operation required by real-time flood forecasting applications. This is touched on in the flow chart of Figure 1 but, in hindsight, we see that this does not emphasize that the steps followed in the flow chart are repeated, in this case hourly, as each new data point arrives. As you point out we had to overcome a number of challenges including maintaining the model state in a situation where the thread running the process could be terminated or powered down. To address this style of operation, the program uses a text file to store and retrieve the necessary state data. These files are read at the beginning of each cycle through the workflow (where one cycle is performed for each model for each new data point). At the end of the cycle the updated state data is stored overwriting the previous version.

We take on board the comments about the lack of clarity in this section and the proposed new version provides a slightly more extended and skillful description together with a summary of the key points described above.

Better description of the gauge sites used by the model

A description of the rain gauge sites forming the input to the outer nodes of the model together with the river level gauge sites forming the inner and terminating nodes is provided in Figure 4. However we will extend the caption to the figure to read:

"**Fig. 4** The configuration of the nodes and gauge sites making up the Eden DBM FEWS catchment model. The square boxes contain the names of the gauge sites providing an input source and are labeled as either rain or level. The rounded boxes represent the individual model nodes. The output from each node is labeled and corresponds to a level gauge site."

In summary

Thank you once again for your useful comments and feedback. We hope we have addressed these issues in this comment (and comments to Reviewer 2) and can go on to provide the revised manuscript.

Interactive comment on Hydrol. Earth Syst. Sci. Discuss., 9, 7271, 2012.

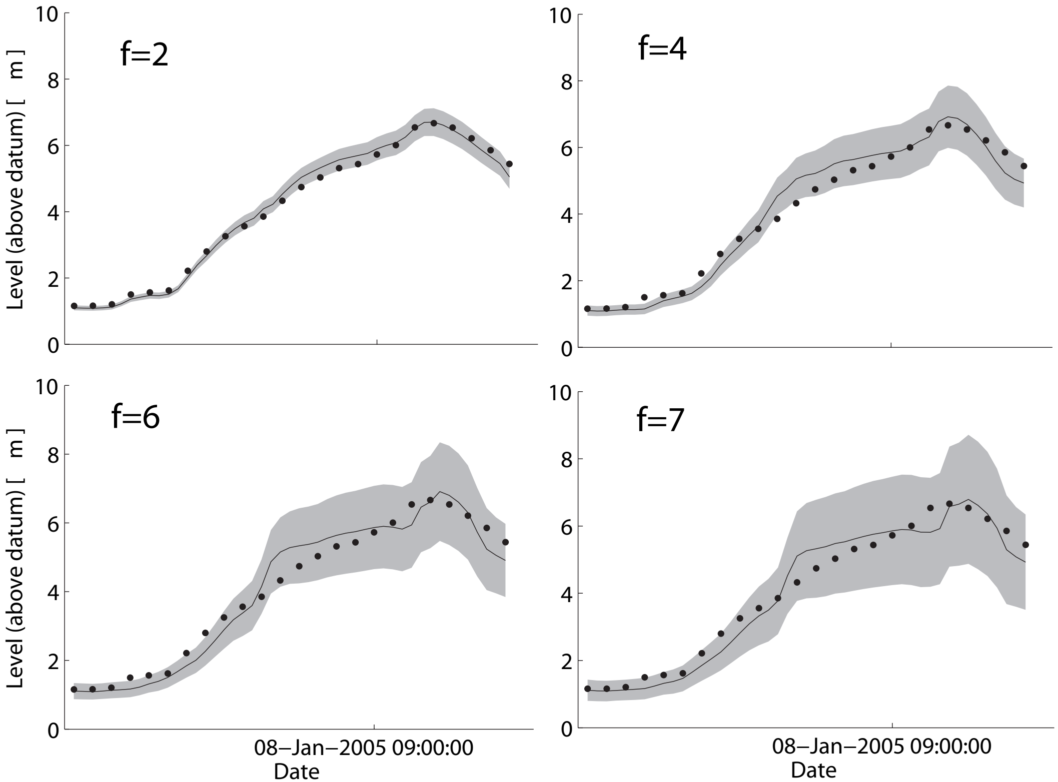

Fig. 1. New version of Figure 5. Caption same as original

HESSD

9, C3955–C3962, 2012

Interactive Comment

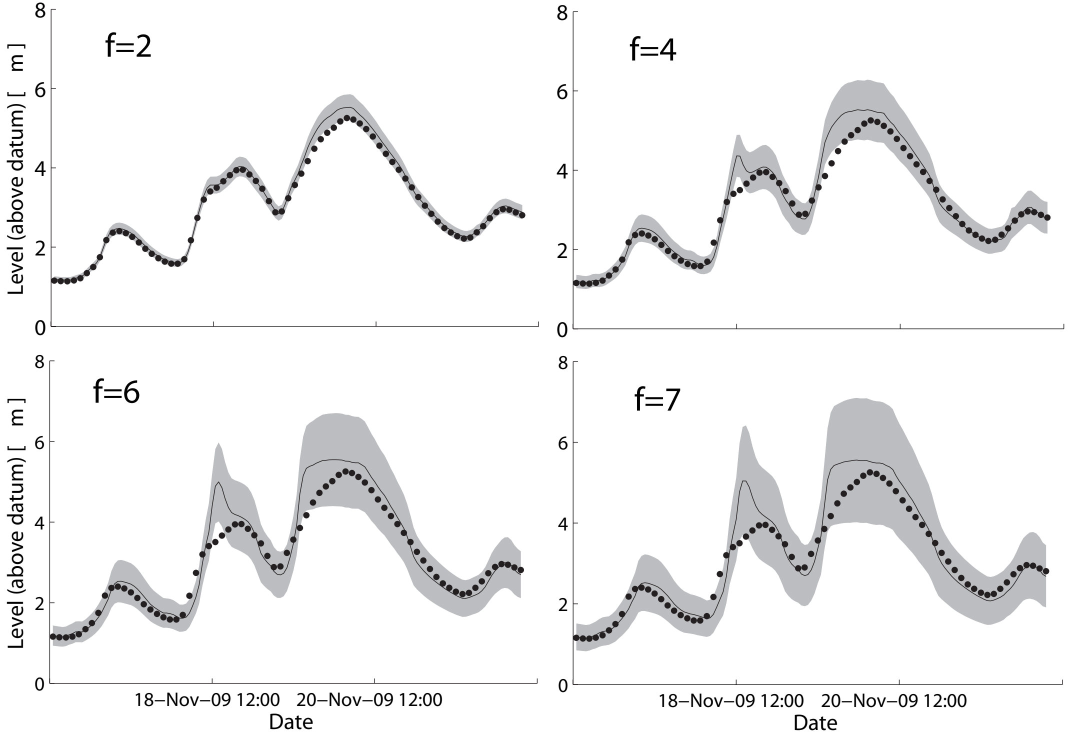

Fig. 2. New version of Figure 6. Caption same as original

HESSD

9, C3955–C3962, 2012