Anonymous Referee #2

Comment: The paper presents a publicly available source code package to assimilate observations into the CLM for state and/or parameter updating. (The source code was effectively available at the time of reviewing this paper, which was greatly appreciated.) While the platform is an applaudable development for this research group, it is oversold in this paper and it not anywhere as well tested and as general as other existing land data assimilation systems. Whereas the authors suggest that their framework would fill a gap in existing land data assimilation systems, I see nothing that would outperform the existing systems or be more advanced than what exists already. Instead, there are a lot of hypothetical statements about what may be possible with the system without any proof or quantitative results. The data assimilation example in this paper is chosen rather poorly: it is fairly simple and illustrates that the assimilation “does nothing”.

My suggestion would be that the authors rethink the paper, and give a more objective perspective on the current state-of-the-art of land data assimilation systems and the position of DasPy present either (A) a complex multi-source and multivariate data assimilation example with a thorough step-by-step validation of the (1) ensemble errors in time and space, (2) data assimilation diagnostics (increments, innovations), and (3) independent validation with in situ observations, or (B) review and show all the tested possibilities of DasPy with quantitative results and highlight how they add value over other existing systems.

Response: Thanks for the suggestions. We want to provide CLM users a small research tool that can be used to test data assimilation, like soil moisture assimilation, land surface temperature assimilation and parameter estimation. DasPy cannot replace other existing data assimilation systems, such as DART and LIS. DasPy is a small research system for the specific CLM community. We are very sorry for the confusion. The manuscript will be revised to make it clear. We will improve the explanation of the data assimilation results following suggestion A, and compare DasPy with other existing systems following suggestion B. Please see the response below.

Detailed comments:

p.7397

Comment: L.5: observation data from multiple state variables observation data related to multiple state variables L.8: most reasonably working land data assimilation systems have a multivariate updating (e.g. a simple case would be multiple layers of soil moisture or temperature) scheme. A multivariate updating scheme is no novelty, but a standard approach. Instead, a multi-source, multiscale data assimilation framework (i.e. simultaneous assimilation of various observation types) would be a step ahead. Is DasPy able to deal with multiscale data assimilation in its current version, i.e. assimilate observations at several resolutions simultaneously, downscale observations, etc.? Please add to the text.

Response: Multiscale data assimilation is still an up-to-date research topic, but a single optimum solution cannot be found to be valid for all problems and improvements are still in discussion. For example, we assimilated remotely sensed soil moisture at a coarse scale (resolution > 25 km) into the high resolution CLM (around 1 km), but we cannot get an obvious improvement. So we are working on the prior downscaling of soil moisture data (Mascaro and Vivoni, 2012; Merlin et al., 2009; Piles et al., 2011; Song et al., 2014). But the prior downscaling analysis should be independent study, and cannot be added into DasPy as well as in this paper. Observations with different characteristics need different downscaling approaches. So DasPy can only be used to assimilate the priorly downscaled coarse resolution observation. We will add more discussion about this part. We hope we can provide this new ability in the next DasPy version. At this moment, only the mature parts are provided.

Mascaro, G., Vivoni, E.R., 2012. Utility of coarse and downscaled soil moisture products at L-band for hydrologic modeling at the catchment scale. Geophys Res Lett, 39(10): n/a-n/a.

Merlin, O., Al Bitar, A., Walker, J.P., Kerr, Y., 2009. A sequential model for disaggregating near-surface soil moisture observations using multi-resolution thermal sensors. Remote Sens Environ, 113(10): 2275-2284.

Piles, M. et al., 2011. Downscaling SMOS-Derived Soil Moisture Using MODIS Visible/Infrared Data. Ieee T Geosci Remote, 49(9): 3156-3166.

Song, C., Jia, L., Menenti, M., 2014. Retrieving High-Resolution Surface Soil Moisture by Downscaling AMSR-E Brightness Temperature Using MODIS LST and NDVI Data. Ieee J-Stars, 7(3): 935-942.

Comment: This is a mix of land data assimilation systems, atmospheric assimilation systems, large operational and small research systems. Where do the authors fit in here? It makes no sense to compare DasPy with something like the operational atmospheric system, nor with a synthetic research system like ATLAS. At the other hand, why are the relevant global land data assimilation systems from ECMWF and NASA missing?

My guess is that DasPy can be compared to e.g. DART or LIS, which are both freely available data assimilation systems and are used extensively for multiple land data assimilation purposes, often at limited domains (like DasPy). The question now is what is it that DasPy offers and DART or LIS does not have? In general, don’t these systems have much more to offer than DasPy and should that not be rightly recognized in a publication?

Many of the systems listed between L.3-L.13 are highly effective large scale, multiscale and multi-source, and multivariate. I can only see the implementation of parameter updating as something that is tested in DasPy and maybe not fully tested in the existing systems (although, several have tested it, e.g. LIS and many research data assimilation systems).

Response: Thanks for the suggestions. We will reorganize the discussion of the different data assimilation systems and focus more strongly on land surface data assimilation frameworks and systems. We will provide for those frameworks a more thorough discussion with the advantages and disadvantages, with a special focus on DART and LIS.

We think that the current DasPy is a small research data assimilation system for the community land model (CLM) only, and not for operational applications. Both DART and LIS are very good data assimilation research toolboxes and the objective of DasPy is not to replace DART and LIS. Our aim is to build a land data assimilation system around CLM. We have tested DasPy in many different data assimilation studies (Han et al., 2012, 2013, 2014, 2015). In the revision, we will make a clear comparison among DART, LIS and DasPy. Sorry for the confusion.

p.7402

Comment: L2-L9: a lot is repeated in the previous page L1-L5, please consolidate

Response: L2-L9 is changed to “The main programming language of DasPy is Python (https://www.python.org/).” L1-L5 is changed to “Python is a free and advanced scripting language, and is also called the glue language as the different programming languages can be integrated easily in Python. DasPy benefits from the powerful Python-based ecosystem of open-source software for mathematics, science and engineering (Fig. 2).”

Comment: L26: “data assimilation is handled separately for each model grid cell”. How does DasPy deal with overlapping areas of influence when spatial extrapolation is performed? Are domain halos implemented? (I only briefly screened the source code and did not immediately find anything that would account for overlapping update areas)

Response: In LETKF, spatial extrapolation is handled by including more surrounding observations for each model grid cell, which is called observation localization. It means one or more observations could be used in assimilation for each model grid cell. The observation localization is done in the LETKF source code. We will clarify this aspect in the revised version of the paper.

Hunt, B. R., Kostelich, E. J., and Szunyogh, I.: Efficient data assimilation for spatiotemporal chaos: A local ensemble transform Kalman filter, Physica D: Nonlinear Phenomena, 230, 112-126, 2007.

Comment: What is the maximum total simulation domain area tested with DasPy?
Response: The largest domain we have tested is 720*360 grid cells. So we have mentioned "DasPy has been evaluated for spatial scales ranging from 100s to 100,000s of grid cells." in section 4.

Comment: Can DasPy run on various grid layouts (e.g. cube sphere, EASE grid,: : :)? It looks like it is hardwired to run on a regular latitude-longitude grid only. Please mention that in the paper.
Response: DasPy can only be run on a regular latitude-longitude grid. This point will be added in the revision.

p.7403
Comment: There is a difference between correlation localization and simulating the spatial error correlations. The text on this page mixes both concepts. L. 6: "spatial correlation characteristics of model state variables and observation data" should be error correlations. This has nothing to do with error localization, this refers to limiting the spatial error correlations in the background error covariance matrix. L. 15: The Gaspari-Cohn method is not used for spatial error characterization, but instead to localize the spatial error correlations. L. 25: please explain all variables; I could not understand this. Eq.(1) has a perfectly identical value at the right hand side for each ensemble member.
Response: We changed the "spatial correlation" to "spatial error correlation". There are two ways to do correlation localization: one is by spatial error correlation in the background error covariance matrix, the other is the spatial error correlation in the observation error covariance matrix which was used extensively in LETKF. Before the correlation localization, we need to define the spatial correlation of the errors. It means the model (e.g. Gaspari-Cohn method) is necessary for spatial error correlation characterization.
There was a mistake in Eq(1), now it has been changed to:

$$\mathrm{X}_i^a = \mathrm{X}_i^a \left(\alpha \frac{\sigma^b - \sigma^a}{\sigma^a} + 1 \right)$$

where X_i^a is the analysis ensemble member i for a given model state, σ^b is the standard deviation on the basis of the model forecast ensemble members and σ^a is the standard deviation on the basis of the model analysis ensemble members X_i^a. α is a tunable parameter between 0 and 1.

p.7404

Comment: Is CLM effectively restarted with a new file after every update? What is the restart frequency? Is the restart frequency flexible for each type of observations? What is meant by “The main disadvantage is the loss of computational efficiency, however, the binary NetCDF file format used by CLM could compensate for loss of computational performance.”? How does the compensation happen?

Response: Yes, CLM is restarted with a new initial file after each update step. The restart frequency depends on the observation frequency (e.g. 6 hours, 3 days, etc.), i.e. it is not constant. The last sentence “however, the binary NetCDF file format used by CLM could compensate for loss of computational performance” is removed in the revision.

p.7405

Comment: Are there any spatial or temporal error correlations in the current DasPy version, and if so, where exactly? L. 18 suggest that there are no spatio-temporal correlations, but p.7403 is all about spatial error correlations and the localization thereof. p. 7406 hints to other data assimilation systems, not something that is implemented in DasPy. If there are no spatial correlations, then I do not understand what p. 7403 wants to convey.

Response: Sorry for the misunderstanding. L. 18 means there is no spatio-temporal correlation in the perturbation of atmospheric forcing data. DasPy however allows to consider spatial and temporal correlations in the atmospheric forcing data. We will clarify this part in the revision. DasPy includes the spatial correlation of observations (P. 7403) for the data assimilation, it is called “observation localization”.

p. 7406

Comment: CMEM = Community Microwave Emission Modelling * Platform *

Response: Changed in the revision

p.7407

Comment: L.24: replace “direct measurements of soil moisture” with “soil moisture retrievals”

Response: Changed in the revision

Comment: How does DasPy deal with biases in each of the various observation types?

Response: The current version of DasPy does not handle observation biases. This could however be included in future developments. We will clarify in the paper that currently observation biases cannot be handled, but could be handled by a priori bias correction methods. The whole issue of bias mitigation is currently a large field of research which cannot be included in this paper (Kumar et al. 2012).

Kumar, S.V., R.H. Reichle, K.W. Harrison, C.D. Peters-Lidard, S. Yatheendradas and J.A. Santanello. 2012. A comparison of methods for a priori bias correction in soil moisture data assimilation. Water Resources Research 48: W03515. doi:Doi 10.1029/2010wr010261.

Comment: Can DasPy deal with spatially variable observation errors? Spatially correlated observation errors? Error correlations between various assimilated observation types? Please comment on observation errors.
Response: We will extend the description in the paper and clarify that spatially variable and correlated observation errors can be handled by DasPy. However, error correlations between various assimilated observation types cannot be handled so far.

p.7408
Comment: L. 5: please add references to support that DasPy can be “easily extended for: :.” (i.e. quantitatively tested cases).
Response: This sentence is changed to “DasPy can be extended for the assimilation of other measurements like land surface fluxes, snow, ground water table and leaf area index. Parameter estimation can be further extended and is not limited to soil properties and leaf area index. More components are under scientific validation, and the current multivariate data assimilation of DasPy is summarized in Fig. 4.” Actually we already implemented the assimilation of snow cover fraction, snow depth, latent heat flux and total water storage change. However, these parts are not validated well, and therefore they are not included in the current version of DasPy. We only provide the components that have been validated extensively.

p.7410
Comment: what is the difference between f(h) and h, or better, what is f(.)?
Response: h (heat flux into the surface soil layer) is the boundary condition of function f(h), which calculate the heat flux in the deep soil layers, so we used f(h) to represent the complex relation.

Comment: what is the difference between lambda and lambda_vap. Please explain all variables.

Response: λ (Wmk^{-1}) is the soil thermal conductivity. λ_{vap} (JKg^{-1}) is the latent heat of vaporization. We will explain all variables.

p.7411

Comment: Please illustrate how is the information in MODIS LST partitioned between Vcmax, Fdrai, Qdrai and all state variables? E.g. what are the update statistics (e.g. standard deviation in increments) to all individual components?

Response: The parameters were updated by multiplying a factor, so the updates were not used on the parameter values directly. The update statistics were not saved, so these parts were not showed. Firstly we generated the ensemble members of soil texture, Vcmax, Fdrai, Qdrai, LAI and atmospheric forcing, and the use of combinations of these ensemble members as model input resulted in ensembles of predicted land surface temperature, soil temperature and soil moisture content. During data assimilation in LETKF, the parameters Vcmax, Fdrai, Qdrai and LAI were updated by LETKF according to the correlation between the parameter ensemble members and land surface temperature.

We will improve the explanation in the manuscript.

Comment: Is MODIS LAI used as input to CLM, and also updated? How does this work? Does DasPy overwrite the MODIS LAI input with new/updated files as CLM is restarted? Is LAI perturbed as described in section 2.4?

Response: The MODIS LAI product was used as input and LAI updated. DasPy updates the CLM input files at each restart step. Section 3.2 shows the details: "In order to represent the uncertainties of these parameters, the default parameter values in CLM were perturbed by multiplying their values with a value sampled from the following uniform distributions: U[0.75,1.25] for $Vcmax$, log2(U[0.75,1.25]) for $Fdrai$ and log10(U[0.75,1.25]) for $Qdrai$. This means that we used the default CLM parameter values as the prior information for parameter estimation. These perturbations were applied only when these parameters were updated together with the states in the data assimilation procedure." We further clarify the raised points in the revised version of the manuscript.

Comment: If MODIS LAI underestimates the true LAI, was any bias estimation turned on in DasPy?

Response: There is no bias estimation provided by DasPy. So we tried to update the underestimated LAI instead.

We will clarify in the paper that currently observation biases cannot be handled, but could be handled by a priori bias correction methods. See also comment above.

Section 3.4

Comment: Please some measure of statistical significance to the results?

Response: We will add the statistical significance in the revision.

Comment: Fig. 7-11: the connector lines have no physical meaning. Perhaps switch the x-axis to show the various validation sites and show the various experiments as different bars or symbols for each experiment?

Response: Thanks for the suggestion, we will change the plots in the revision.