General Comments

Tang et al described a new developed geospatial probabilistic estimation package (GPEP) to generate ensemble meteorological datasets based on station observations. They demonstrated good performance of GPEP with some examples. Compared to Gridded Meteorological Ensemble Tool (GMET), GPEP has several improvements, such as multiple selection of spatial interpolation methods, additional meteorological variables, user defined inputs, and etc. However, the computational performance of GPEP can be worse than GMET since GPEP is developed in Python. The computational performance is improved by using multiprocessing package, and they show speed of the simulation is significantly increased as more cores are used. GPEP represents a useful tool for the Earth science community. Specifically, the gridded meteorological variables that generated by GPEP can be used as forcing data for Earth System Models, hydrological models, especially for ungagged regions. The ensemble estimations can be further used for assessing the uncertainty caused by meteorological forcings in the simulation. Although I think this study could be a significant contribution to ensemble geophysical datasets estimation, it can be further improved before publication in Geoscientific Model Development. Please find my major concerns and specific comments in the following.

Response: Thank you for the comprehensive review and insightful feedback on our manuscript. We have addressed each of your comments in detail below and revised the manuscript accordingly.

Major comments:

1. GMEP considers the spatial correlation and correlation between temperature and precipitation. As more variables can be processed and generated in GPEP, are the intercorrelation among variables considered? For example, in the application of Earth System Model, precipitation, humidity, radiation, and temperature are needed. Those variables are correlated to each other in time and space. Is it possible for GPEP to generate those variables together? In addition, how other variables are generated in GPEP is not described. Eq (1) – Eq (4) describes how temperature and precipitation are generated in GMEP. Can the same equations be used for any other variables?

Response: We believe you are referring to GMET when you mention GMEP. The problem you point out is valuable. In the original GMET methodology, the dependence relationship between precipitation and daily temperature range is defined via their cross correlation, as depicted in Equation (1). GPEP extends this capability, allowing users to specify dependencies between any pair of variables, such as precipitation versus temperature range, or precipitation versus humidity, or temperature versus radiation, through the “linkvar” parameter in the configuration file.

However, the current implementation does not handle intercorrelation relationships among more than two variables. For example, for precipitation, humidity, radiation, and temperature, Equation (1) cannot consider how each variable is correlated to the other variables. A workaround is to decompose the intercorrelation into multiple binary dependencies. For example, we can define three dependence relationships here: (1) humidity depends on temperature, (2) precipitation depends on temperature, and (3) temperature depends on radiation. GPEP can address this by first

generating radiation random fields, followed by temperature random fields based on the radiation, and subsequently, humidity and precipitation random fields based on the temperature. However, this solution is less efficient than jointly estimating multivariate cross-correlation or covariation matrices as part of a generalized multiple linear regression approach.

Generalizing the current bivariate cross-correlation implementation is a high priority for future development, but will require some further code development effort. We are also interested in exploring alternatives such as using copula functions, which may be facilitated by providing a python code-base which links to more existing packages than Fortran. In any case, we have added a new comment in the Discussion section of revised manuscript recognizing this current limitation as a future research direction.

About your second question, the answer is that other variables can also be generated using Eq (1) – Eq (4). The probabilistic estimation framework is applicable to any variable following a normal distribution. For variables that deviate from this distribution, such as precipitation or snow water equivalent, a transformation is necessitated, as well as (depending on application) the treatment of the variable as intermittent, with an associated probability of event. GPEP users have the flexibility to adjust the exponential factor of the box-cox transformation in the configuration file to ensure the transformation is suitable for their target variables, and in the future will have a broader range of transformation options. We have explained this in Section 2.

2. What is the roadblock for running GMEP for the High-resolution meteorological forcing ensemble generation? It will be interesting to see if GPEP has better or similar performance than GMEP in such application.

Response: This question is partly answered in Section 4.1, which compares GMET and GPEP in a 1/16th degree resolution daily meteorological ensemble generation for California. While the term "high resolution" might be subjective and vary among individuals, the mesocale 1/16th degree results indicate that when GPEP is configured to mirror GMET settings, both tools can yield nearly identical estimates.

Roadblocks for large-domain high-resolution estimation currently the computational efficiency of Python, although parallel computation can alleviate this problem (Figure 2). Other options include algorithm optimization, hybrid programming, and shift development from CPU-based computing toward using GPUs. We have added discussion in Section 5. Another issue that may be particular to applications in which the predictors are highly variable at granular, high resolution scales, is that they can require a greater degree of extrapolation, which may give rise to unrealistic behavior at extreme locations from a predictor or predictand perspective. For example, a terrain-based predictor at fine resolutions can have a much larger range (spanning elevations perhaps 1000-5000m) while the training data is unlikely to fully represent observations across this full range. Co-author Wood has run GMET experimentally at resolutions up to 250m, and noted some issues in this regard. Solutions may include different forms of regression (such at gaussian process regression) that recognize variable data support, or certain machine learning approaches, and adjust the uncertainty estimation accordingly.

3. There are other 1km reanalysis meteorological datasets that can be used for benchmarking GPEP's simulation for the high-resolution meteorological forcing ensemble generation example (i.e., Sec 4.2). For example, Daily Surface Weather Data on a 1-km Grid for North America (Daymet). Is GPEP better than existing high-resolution reanalysis dataset for reproducing the meteorological variables at the station locations? How is the spatial pattern compared to such high-resolution reanalysis dataset. I think by adding such benchmark and evaluation can give the readers/users more confidence on the application of GPEP.

Response: A short answer is that we haven't had the scope of effort to include a detailed Daymet (or other dataset comparison), although a new study by a PhD student of Co-author Wood intercomparing GMET/GPEP with 5-6 other dataset options is underway. It is an excellent topic, but not one we can delve deeply into here. We agree that adding a high-resolution dataset as the benchmark would be an interesting comparison point for readers, although we do not know the absolute quality of Daymet, and the GPEP case study is not a production quality dataset. There are several reasons that we do not include the comparison in the manuscript.

(1) A prior paper by Henn et al. (2018) assessed the differences in gridded precipitation datasets in complex terrain (e.g., the western CONUS), including the GMET-based CONUS dataset (Newman et al., 2015), Daymet, and some other widely used datasets. In this manuscript, we have demonstrated that GPEP can reproduce the outputs of GMET with the same settings used in Newman et al. (2015). We have touched on the comparison results from Henn et al. (2018) in the revised manuscript.

(2) GPEP is a tool with myriad configuration choices for estimation, rather than a program with a fixed setup to produce only meteorological datasets. Numerous datasets can be generated by changing parameters in the configuration file, while exploring optimal configurations or comparing a specific configuration to existing datasets has gone beyond the scope of this software paper. Users need to fine-tune GPEP's settings to achieve optimal performance tailored to specific regions or variables, and we have mainly provided case studies to illustrate the working of GMET without tailoring them as standalone, high-quality datasets for broader use (like Daymet).

(3) Cross-validation capabilities: A notable feature of GPEP is its ability to offer cross-validation results, as illustrated in the manuscript. This allows users to gain an objective insight into GPEP's statistical accuracy or reliability without comparing it to other datasets – essentially, it quantifies its own ability to estimate out of sample observations. Evaluating through comparison to existing datasets like Daymet can be challenging since they often integrate ground station data into their production.

We have added discussions in Section 4.1 and Section 5.

Reference:

Henn, B., Newman, A. J., Livneh, B., Daly, C., & Lundquist, J. D. (2018). An assessment of differences in gridded precipitation datasets in complex terrain. Journal of Hydrology, 556, 1205–1219. https://doi.org/10.1016/j.jhydrol.2017.03.008

Newman, A. J., Clark, M. P., Craig, J., Nijssen, B., Wood, A., Gutmann, E., et al. (2015). Gridded Ensemble Precipitation and Temperature Estimates for the Contiguous United States. Journal of Hydrometeorology, 16(6), 2481–2500. https://doi.org/10.1175/JHM-D-15-0026.1

4. The authors argued GPEP features multiple selection of spatial interpolation method. But still the locally weighted linear/logistic regression were used in all the demonstrations. I think it is necessary for the author to show the application of other spatial interpolation method. Specifically, will using supervised learning method for the spatial interpolation can improve the performance in Figure 4 (the high-resolution meteorological forcing ensemble generation)?

Response: Thanks for the suggestion, and we agree that it is important to demonstrate machine learning-based experiments in the manuscript. In Figure 2, we have shown the comparison of the parallel efficiency between locally weighted regression and random forest. In light of your feedback, we have further enriched our manuscript by incorporating experiments that utilize supervised learning techniques and compare machine learning's performance with that of the locally weighted regression in Figure 5 (formerly Figure 4).

We added a new figure (i.e., Figure 6) in Section 4.2 to compare the performance of random forest estimates and locally weighted regression. However, please note that this just demonstrates an application of one machine learning method. Exploring the superiority of machine learning versus locally weighted regression would necessitate a comprehensive exploration of hyperparameters and feature combinations to optimize machine learning performance. Conclusions may also vary for different regions. The GPEP software or the manuscript cannot provide a more quantitative and comprehensive conclusion here.

5. Does GPEP only accept gauge data as input? Can we use gridded reanalysis dataset as input and output at higher spatial resolution? In addition, can GPEP generate the meteorological datasets at sub-daily scale? This can be useful for using GPEP to generate forcings for models, as some models requires sub-daily meteorological forcings.

Response: This is definitely possible with GPEP (and GMET v2.0, albeit with more pre-processing required for the latter). A major goal of allowing for dynamic predictors in both software was to enable a fusion of station and other forms of information, such as gridded analyses (reanalysis, NWP, satellite, radar). Hence GPEP can work with station data and also incorporate reanalysis datasets as dynamic predictors for spatial estimation. If users wish to use reanalysis data as direct inputs, they can reformat these datasets by treating each grid cell as an individual station. This involves reshaping the reanalysis grids from a (time, x, y) structure to a (time, x·y) format. Once reformatted, these can be fed into GPEP as standard inputs, allowing users to generate outputs at their desired spatial resolution.

About the second question, GPEP can generate forcings on any temporal scale, including sub-daily intervals as long as sub-daily input data are fed into the package.

We have explained more about the two questions in Section 5.

Specific Comments

Line 32: In my experience, reanalysis dataset commonly has better temporal coverage than the station observation. Please cite relevant reference to support this statement.

Response: We intended to convey that ground stations offer the longest meteorological observations, while reanalysis datasets are estimates. To avoid confusion, we have removed the word "longest".

Line 46: Full name for HadCRUT4?

Response: We have included the full name, i.e., Hadley Centre/Climate Research Unit Temperature version 4.

Line 88 and Line 89: What are continental EMDNA and global EM-Earth?

Response: The full names and references of the two datasets were mentioned in Line 49 and Line 50.

Line 98: Is temperature range = maximum temperature – minimum temperature in the day?

Response: Yes, the temperature range is defined as the difference between the maximum and minimum temperatures of the day. We have clarified is in the revised manuscript.

Line 235: I wonder the computational performance for large scale application, for example, continental or global simulation.

Response: The speed of GPEP is influenced by many factors, such as regression parameters, chosen resolution, and domain size. Our upper Colorado test case, with a 0.01-degree resolution, has 275×275 grids. In comparison, the North American Land Data Assimilation System (NLDAS) uses a 0.125-degree resolution, resulting in 224×464 grids. The upper Colorado case, having about 73% of NLDAS's grid count, can already provide a benchmark for larger domain applications. In the upper Colorado test case, we utilized 36 CPUs on the Casper HPC at NCAR. It took 5,560 seconds to produce regression estimates and 36 ensemble members for the year 2013. Note that this duration does not account for the one-time generation of prior files, such as indices for neighboring stations and the spatial correlation structure.

Additionally, while GPEP supports locally weighted regression, it also offers faster global machine learning regression methods. Direct comparisons with the GMET package can be challenging due to methodological differences.

We have added some further explanations for Figure 2 and in Section 4.2 so users can have a basic estimate of the computation time for a similar application.

Section 4.1: I suppose the same algorithm was adopted in GPEP. So, is the difference caused by the random seed used in GMET? Can the authors further explain the attribution for the differences?

Response: Indeed, both GPEP and GMET employ the same locally weighted regression algorithm. The differences observed aren't due to the random seed, as Section 4.1 focuses on deterministic estimates. While GPEP is Python-based and GMET is Fortran-based, their algorithmic performance is consistent. Differences in their cross-validation approaches can also lead to differences in their deterministic and uncertainty estimation. We have conducted independent numerical comparisons (i.e., extracting regression functions from GPEP and GMET packages) to confirm this. In Figure 3 (formerly Figure 2), most estimates from GPEP and GMET are the same. But minor discrepancies, especially in the probability of precipitation, also come from slight numerical differences in data inputs, attributed to differences in double precision or single precision in GPEP and GMET codes. These minor variations can be magnified during iterative processes of logistic regression. We have elaborated on this in our revised manuscript.

Figure 2: Does subplot (a) represent the mean precipitation of the simulation period, or all the daily precipitations within the simulation period?

Response: It represents all the daily precipitations within the simulation period. Specifically, each point corresponds to the values for a single grid at a particular time step. We have clarified this in the figure caption.

Figure 3: What is the source and spatial resolution of the elevation data? Does GPEP estimate the south-north and west-east slopes internally, or does the user need to preprocess it? It will be useful for the authors to describe how the slopes were calculated.

Response: The elevation data is from SRTM with an original resolution of 3 arc-seconds. This data was subsequently resampled to a 0.01-degree resolution for the test case. GPEP does not automatically compute the south-north and west-east slopes. Instead, users are responsible for determining the predictors they wish to use in their studies. For the test case presented, the south-north and west-east slopes were derived from a smoothed Digital Elevation Model (DEM) to minimize high-resolution noise and emphasize broader topographic influences. Further details on the slope data preparation have been provided in the revised manuscript. The prior GMET applications have led to the development of a range of grid processing scripts (unpublished so far) to prepare such inputs.

Line 317: Could the higher performance in the flat eastern areas be due to there being more stations in this region? This raises the question: what is the minimum number of stations needed by GPEP for good performance? What is the performance of GPEP in data-sparse regions? I am not asking the authors to run additional simulations for this question; some discussions and perspective from the authors will be very helpful for the readers.

Response: Indeed, the higher density of stations in the flat eastern areas likely contributes to the enhanced performance observed in that region. We have elaborated on this in the revised manuscript.

The question of the minimum number of stations required for GPEP to function optimally is multifaceted. The optimal station density can vary based on environmental conditions, targeted

meteorological variables, and methodology choices. For instance, certain machine learning techniques could be less sensitive to station densities compared to traditional interpolation methods. We have discussed more about this issue in the revised manuscript.

Regarding the performance of GPEP in data-sparse regions, it is hard to predict because of the abovementioned reasons. However, the flexible configurations in GPEP offer users the chance to explore optimal regional performance compared to using public datasets or tools with limited choices. We expect that like all statistical modeling problems, GPEP performance will be undermined by data sparsity, and this was another reason for including the ability to use dynamic gridded predictors, such as reanalysis or NWP – in regions of sparse in situ data, one would do no worse than the gridded background fields, ideally.

We have added discussions in Section 5.

Line 322 – Line 325: Can the authors quantify the ensemble spread? For example, the spatial correlation between any two ensemble members.

Response: To quantify the ensemble spread, one common approach is to use the standard deviation. We have incorporated ensemble spread maps in Figure 7 to provide a clearer visualization of this spread. The ensemble spread of temperature shows that during that period, the uncertainty mainly occurs at stations located in the southern part of the study area.

Line 326: Is there a station located inside the area selected by Figure 6? If so, how does the simulation ensemble compare to the observation?

Response: Yes, some stations are situated within the area highlighted in Figure 7 (formerly Figure 6). We did compare ensemble observations to station observations. The ensemble mean generally agrees well with station observations, which is as expected because the comparison is not independent due to the integration of station data in ensemble estimates. According to the probabilistic method, the probabilistic estimates should fluctuate around the true value if the deterministic regression is accurate enough.