

General comments

The manuscript “*Open-source tools for processing opportunistic rainfall sensor data: An overview of existing tools and the new OpenSense software packages poligrain, pypwsqc and mergeplg*” by C. Chwala et al. presents a set of open-source software tools for processing and visualizing different types of opportunistic precipitation data.

The 3 types of opportunistic sensors (OS) dealt with are i) commercial microwave links from mobile phone networks, also known as CMLs, ii) satellite-ground microwave links -SMLs- from TV broadcast networks and iii) personal weather stations, PWS.

The objective of the paper and of the proposed suite of packages is to help widening the community of users that will be able to process opportunistic sensors to obtain rainfall estimates, for many applications. The paper is somehow a cross between a review paper (with many scientific references to scientific and methodological papers on opportunistic sensors) and a technical note with description of open-source packages. This form is unusual and leads to a long paper, but in my opinion, the structure and content of the paper are well suited to the objective: promoting and facilitating the use of OS by a range of users with different profiles and levels of expertise.

The task undertaken by the authors was both necessary and challenging, and I congratulate them for this initiative.

As discussed in the introduction and in the cited papers, the usefulness of opportunistic sensors as a way to improve the global observation system for rainfall has been proved by many studies. OS is now being integrated in the operational rainfall products of a few meteorological services – mainly in Europe. However, the deployment of these solutions in the Global South and in data scarce regions where OS could be a real game changer remains a challenge because most of the expertise on subject is concentrated in the richest countries. Publication of the paper in a scientific journal like HESS and the companions GITHUB repositories, will help raising awareness on OS, while providing the tools for interested parties to gain theoretical and practical expertise on the topic.

I believe the paper should be published when a few minor comments detailed below have been accounted for.

Specific comments

Section 2 and 3 :

First, I would like to mention that unlike the anonymous reviewer who sent a comment, I find the present structure with extended sections 2 and 3 discussing the existing processing and packages very relevant. It is important for potential users to understand the history and genesis of the OpenSense community, and to have access to the original methods and packages. It is also a tribute to the people who have contributed along the years to the collective effort.

I also agree with authors' choice to include SML, PWS and CML in the paper, even though CML processing is more mature than the 2 others. It will encourage users/experts from the 3 communities to share methods.

In section 3

->I suggest adding to Table 2 a line concerning the type of data each package is able to handle : Specific time step ? able to handle mean and current measurement in addition to Min and Max ?

Section 4 :

Section 4, presenting the new vision and packages could be improved.

The 3 presented packages, poligrain, pypwsqc and mergeplg have not reached the same level of maturity and readiness. The reader discovers these differences when exploring the corresponding github repository, I believe it would be better to make it clear in the paper.

->I suggest changing the titles of sections of 4.3 and 4.4 to

pypwsqc: A compilation of PWS QC methods – package in development

mergeplg: Merging methods for point, line and grid data - package in development

-While the documentation in the poligrain github provides an overview of the package workflow and information on the methods, the description of pypwsqc and mergeplg, is very shallow in the paper and non-existent in the github. For pypwsqc, the interested user needs to navigate to the previous version (<https://github.com/AbbasElHachem/pws-pyqc>) in order to find some description of the work flow and methods applied.

->The authors should provide more technical details in 4.3 and 4.4 on the methods that will be implemented, and the choice of parameters and values (threshold etc..).

->For instance, how do the merging methods will account for uncertainty in the various sources of data ? Will Kalman filters be used ? if Kriging is used how will the variogram be computed ? etc.. etc...

-> A discussion of the implemented methods would certainly help the reader choose the appropriate tool for processing their data. Do all the methods mentioned yield comparable results? Are some better suited to a particular type or quality of data than others? How can the reader make an informed choice when processing their data?

-> One of the major beneficiaries of opportunistic data (such as CMLs) are African meteorological services, which generally have sparse observation networks despite a context of tropical rainfall. Since these countries generally do not have radar, the authors could discuss the possibilities of merging CML rainfall data with other available data sources (satellites or rain gauges).

Guidelines for users – default implementation

As the paper is aimed as a wide public, including non-experts of OS and the package integrates multiple options for processing steps, it could be interesting to provide the reader with suggestion of a 'default' pipeline, using the most simple or robust methods among the choices proposed in the packages. Choice of baseline method etc...

Also recommendation about which processing options are most suited in different context (which interpolation method is recommended for which network density ? which baseline method for which time step ? for which quantization step ?)

A synthetic table or workflow to guide the user in its choice would be helpful.

Guidelines for potential contributors

At the opposite end of the reader spectrum, it would be interesting to provide guidelines to experts who might want to contribute to the packages.

Typographical comments

Figure 1, left panel, on the plot title “Typical reseach software...”, the letter “r” is omitted. Please write “research” instead of “reseach”.

Lines 234–235: “... (De Vos et al., 2019).” Please correct the citation; perhaps it should be De Vos et al., 2019a?

Line 426: “... station-outlier-filter from de Vos et al. (2019a), ...” Insert a blank space between “from” and “de” in “de Vos et al. (2019a)”