Review on egusphere-2025-5438 by Chwala and colleagues

General comments

In their 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"*, C. Chwala and colleagues present a set of software tools for processing of opportunistic precipitation sensors.

I appreciate the achievements that were accomplished in OpenSense and congratulate the authors for their contribution. The presented packages will certainly make an impact in the community and it is worth presenting these accomplishments in HESS in order to support the establishment of standardized data, metadata, and processing workflows as well as collaboration in the future development of the presented tools. My compliments specifically to the polygrain package which looks great and provides excellent documentation.

Having said that, I feel that the paper still needs some work to achieve its purpose. To summarize my main impression: the manuscript could be shortened dramatically - let's say by 30-50% - without losing much informational value. Please let me explain why.

In my view, the paper has two major merits: first, providing the potential and limitations of opportunistic rainfall sensors (introduction) together with a review of data processing requirements (section 2), and, second, by presenting the new tools developed in the OpenSense context (poligrain, pypwsqc and mergeplg) together with examples (section 4). Meanwhile, most of section 3 could be dropped. While I understand the motivation of section 3, it very much impairs the flow of the paper. If you think that specific packages presented in section 3 integrate well with your OpenSense ecosystem, so if it is more than legacy, and if it actually qualifies as a package (a package, in my view, also includes documentation and support), then present it along with the packages presented in section 4. Otherwise, a brief reference will suffice. So altogether, the idea of presenting methods from *before* and *after* OpenSense is not really helpful, in my opinion. It makes it difficult for the reader to keep track of what is relevant. Please focus on the ecosystem that you consider essential, and invest the resources of the paper to present it in a concise way. Example: if pypwsqc is designed to override / replace previous packages on PWS QC, there is no need to present these previous packages in much detail.

But also sections 1, 2, 4 and 5 could be substantially shortened. In the end, a paper such as this one cannot be comprehensive in terms of technical and scientific detail, and any attempt to strive for such comprehensiveness is likely to break the scope of a scientific paper. In my view, there is a tendency to overload this paper with details about steps that are admittedly laborious and important, but which are also trivial and obvious - such as associating time series with sensor metadata. On many occasions, the paper includes anecdotal, circumstantial or historical details (originally developed at…, for the first years…, meant to be…, over the years…) that might be dear to the authors, but which in the end reduce readability and conciseness. So in terms of length or level of detail, my recommendation

would be that the authors should aim at something like a technical note, although, in my view, it is not important to pin such a label to a paper - I just want to provide an example for a format that is more geared towards the subject.

Another recommendation is that the authors should consider establishing a platform that actually brings the available (and relevant) software packages in the field together in terms of presentation. The advantage of such a platform is that its content can change over time so it will likely not become outdated as quickly as a paper that presents specific packages in quite some detail. Maybe the authors could use the openradar site (https://openradarscience.org/) as inspiration for such an endeavour - no need to be fancy here. You could also follow a similar concept as openradar in having core projects (from OpenSense) and affiliated projects (from outside OpenSense, but still active and part of the ecosystem). The GitHub organisation https://github.com/OpenSenseAction/ provides an excellent basis to start with.

Another fundamental issue is that PWS and CLM/SML are different in many respects. The main common feature is actually that they were all subject of investigation in the OpenSense project. Personally, however, I think that specific project contexts are not so much of interest to the audience of a scientific paper (you might even consider dropping section 1.4). This issue becomes repeatedly acute throughout the paper, when specific facts apply to one part, but not to the other. This is obvious with regard to the spatial scale of the observation, but also applies to issues such as availability. I do not have a good idea how to address this issue. Maybe the authors should put, from the very beginning, more emphasis on the issue of integration: that one type of opportunistic sensor will rarely be applied on its own, but they should be integrated with other sensors. And for this integration it is vital to treat data at different spatial scales. This is why polygrain is so valuable and it is a good choice to consider it the basis of all your efforts.

Altogether, I am confident that the authors will find an adequate way to revise the paper, to find an improved balance of what to present in and outside the paper (e.g. on a community platform).

Specific comments

- I recommend removing SML from the main scope of the paper. There is no significant code or analysis with regard to SML involved, so it is hard to justify it being so prominently addressed. Unless you bring in substantial results for SML, it will suffice to mention that many methods of polygrain and CML-related packages might be useful in SML processing and analysis.
- In comparison to polygrain, mergeplg seems to be rather immature at the time being. In particular, documentation is scarce (except one jupyter notebook). From a user perspective, insufficient documentation is *the* single most important barrier. I thus recommend completing the documentation of mergeplg before revising the paper.
- Title: Mentioning the package names in the title is not helpful. Please consider a new title, e.g. just drop everything *after* the colon.

- ll. 43 ff.: Bear with me, but please explain briefly the background of these satellites? Are these actually satellites for TV broadcasting?
- ll. 57 ff.: *“From a methodological point of view, the use of OS data poses several challenges primarily due to the lack of control over the sensor installation and maintenance, which often lead to substantial measurement error”* - please elaborate further. I understand how this is an issue with PWS, but why with CML/SML?
- All section 1.3: For me, the distinction between "methodological” and “technological” issues is unclear.
- ll. 67 ff.: *“ultimately help address the organizational and commercial barrier”* - how so?
- ll. 70 ff.: I appreciate that you provide background on each letter of FAIR, however, what about the *“R”*?
- Fig. 1: this is a very heavy figure that contains very little information and adds little to the understanding of the problem
- ll. 103-106 (*“Before [...] conventions.”*): typical example of information that could be dropped, as this is self-evident (although it might be difficult)
- l. 110: What is a “simple filter”?
- l. 110: *“MLs with specific metadata”* - I would suggest “MLs with specific attributes” instead
- l. 112: *“too long”* - on what basis is such a decision made?
- Fig. 2: I do not find the figure helpful in understanding the manuscript. It would be more helpful to support section 2.1 by a figure that shows an exemplary or idealised time series and the effects of the different processing steps (filtering, event detection, baseline, …).
- ll. 139 ff.: *"However, in the SML case, a constant (though unknown) value is assumed for the TSL.”* - please explain.
- ll. 140 ff.: *“Furthermore, the raw RSL data can be obtained either from diagnostic measurements in commercial satellite broadcast receivers, or by purposely developed devices.”* - is that important?
- ll. 149-151: *“Also for SML [...] conventions.”* - drop?
- l. 158: *“glitches or step-like variations”* - this is a typical example where a schematic figure with exemplary time series would be useful
- ll. 159-160: what do you mean by *“cross-check”*?
- ll. 165 ff.: *“Machine Learning (ML) and other AI-based methods can be used for wet/dry classification”* - references?
- l. 177: no attenuation assumed to take place at all in the solid phase above 0°C isotherm?
- ll. 204-215: Many QC processing steps for PWS involve the comparison with other reliable sensors (rain gauges, weather radar) which limits the application in so-called data scarce regions. As this is a major PWS application scenario, these limitations should be discussed. In the end, I have the feeling that data-rich regions will benefit most while data-scarce regions are used as a justification mainly (correct me if I’m wrong…well, this, of course, applies to the entire review).
- ll. 204-210: In a similar vein, spatial consistency tests will be difficult in data-scarce regions. Moreover, they are highly subjective, as they imply that plausible patterns are essentially coherent in space, an assumption that will fail in case of convective heavy rainfall events - which is a situation in which we hope the PWS to actually unfold their value. Please discuss briefly or refer to corresponding studies.

- l. 221-223: *"Since there is no open-source software package for SML data processing yet, we refer to small existing open-source code snippets from that domain."* - In my view, this is not enough. If no OSS packages exist, a package should be created that includes the corresponding functionality. In this context, the author should reflect the significance of the term "package". In my view, the term not only implies support of installation, distribution, and management, but also documentation and user support. If the aim of this study is to establish an ecosystem of packages to standardize specific workflows, snippets will not do.
- ll. 288-293: example of text that could be discarded without losing relevant information
- ll. 298 ff: *"The overall assumption of the method is that, while individual PWSs may be incorrect, by evaluating the median of nearby PWSs, one can reliably determine rainfall amounts and dynamics."* - I would argue that this is not the case in densely settled urban settings with high buildings in which the majority of PWS might be affected by shielding effects.
- Section 3.3: I do not know what to make of this. Essentially, no SML packages exist, yet. Why include this in the paper?
- Section 3.4: The sandbox is an excellent idea, but only in case it is actively supported, not if it is pure legacy. In that case, it might stand at the end and include all packages, also the new OpenSense packages.
- ll. 367-372: not clear how these packages are used, except for "inspiration"
- Section 4.4: please briefly discuss the relationship/interoperability to/with other merging implementations such as in wradlib's adjust-module.
- Section 4.1 or 4.2: Please discuss how the poligrain approach related to xradar https://docs.openradarscience.org/projects/xradar
- l. 449 ff.: please provide more background on the OpenMRG and OpenRainER datasets and their role in the presented framework
- l. 471: *"Ideally the radar-based wet-dry classification is combined with a CML-based one, but this is still an open research topic."* - why is that? Why not contrast a CML-based classification from pycomlink with a radar-based one?
- l. 474: *"similar to what the basic example provided with pycomlink does"* - what does that mean?
- Fig. 7: Why not show one scatter plot of radar vs. gauges for both scenarios instead of the difference plots (which are not very illustrative)?
- Conclusions section could also be shortened.

Technical comments

- Title of section 1.2: please introduce acronyms before using them in subsection titles
- L. 76: *"rainfall estimation"*, not observation
- L. 78: "open-source implementation" instead of *"open-sourcing"*
- L. 79: *"However, it is only through the coordinated work plan and incentives provided by OpenSense that a solid foundation for continued progress has been established"* - sounds trivial to me: this applies to any project context (although I would be curious about the "incentives")
- Inline citations are repeatedly formatted incorrectly, see e.g. l. 111: it should be "[...] see e.g. Overeem et al. (2016a) and Blettner et al. (2023)" or "[...] (see e.g. Overeem

et al., 2016a; Blettner et al., 2023)". This can be found on several occasions throughout the paper.

- l. 123: "leads" instead of *"lead"*
- l. 127: no comma after *"noted"*
- l. 129: delete *"going to"*
- l. 133: "points" instead of *"point"*
- l. 134: "methods" instead of *"method"*
- At this point, I stop correcting typos and other such mistakes. Please carefully check the manuscript w.r.t. punctuation, syntax, language etc.
- l. 143: explain unit dBm
- l. 146: *"wireless network operator"* - "mobile network" instead?
- Phrases like "It is worthwhile to remark" or "it should be noted" and the like are used a bit too often, at least for my taste, with no specific effect except that it inflates the text.
- ll. 171 ff. : please consider using "offset dish antenna" instead of *"offset reflector"*
- l. 243: "*KIT"* - explain acronym
- l. 243 ff.: while pycomlink implies that this is a Python package, this should be spelled out explicitly - the only explicit reference in this paragraph is to Matlab
- l. 265: explain acronyms (SNMP, DAQ)
- l. 279: explain acronyms (IDW, GMZ)
- At this point, I stopped tracking unexplained acronyms - please address this issue throughout the manuscript (many occasions left!)
- l. 280: *"rain field simulator"* - wouldn't the term be "weather generator"?
- l. 282: *"their attention"* - I guess it's "attenuation"? To what refers "their"? And the term "this grew PyNNcml" sounds awkward.
- l. 308: *"met"* should be replaced by "meteorological"
- l. 321: "a", not *"an"*
- l. 382 ff.: " i.e. the interplay of the different new and existing software packages is envisioned" - missing "how"?
- l. 288: *"which live one layer"* - please rephrase
- l. 392: *"leverage the fact"* - what does that mean?
- l. 400 ff.: *"It was built from scratch in early 2024 as a result of the assessment of the current processing packages to centralize common functionalities that individual processing software packages might share. From the start we followed modern best-practices of software development."* - typical example of text that could be dropped without replacement. The following text *"We based our repository on GitHub on a cookie-cutter template for Python packages that enables strict style check and static code quality analysis using ruff, continuous integration with running unittest and testing the Jupyter notebooks, automated generation of documentation and continuous delivery of package releases on pypi. In addition, a conda-forge build pipeline was set up."* could also be dropped, in my view, as it is unspecific to the subject and meanwhile common practice.
- Fig. 3: font is very small, please rearrange to improve readability
- Fig. 4: How does *"plg.example_data.load_openmrg(subset="8d")"* control which subset is used?
- Fig. 5: colorbars lack unit, and there is no explanation of what the plots actually show. I also suggest including timestamps in the plots and zoom in - otherwise, there is not much to see in terms of comparison.

- Fig. 6: same as with 5 - please extend caption to explain the figure elements. Briefly explain the indicator metric. What does the x axis of the top level plot show?
- l. 488: Meaning/relevance of “total loss” has not been explained before. Explain what is remarkable about the TL time series.
- ll. 491: “simple IDW-based additive radar adjustment” - explain in brief
- l. 528: footprint