Dear Stephen,

I hereby submit a revised version of my manuscript “PyOcto: A high-throughput seismic phase associator” to Seismica. Thank you for the handling of the manuscript and for keeping me updated throughout the process.

In my revision, I considered all the helpful and constructive points raised by the reviewers. Please find attached a point-by-point response (written in blue) to the comments of the reviewers (written in black) with the revised article in two versions, one with changes tracked and one without. The line numbers in the response letter refer to the version with highlighted changes.

I thank the editor and the two reviewers for pointing out typos and recommending improvements to formulations. I have implemented these changes but omit them from the responses below to keep the response letter more compact.

I hope the manuscript now meets the high standards associated with Seismica.

Best regards,

Jannes Münchmeyer

Editor comments

I feel that the Introduction would read more coherently if you merged the “Background” section into the Introduction. I think discussing these existing associator approaches would then allow the motivations for designing PyOcto and the paper’s objectives to be more coherently defined. Also, can you comment on the various pros and cons of the various existing methods?

> Thank you for the suggestion. Upon reading the sections another time, I’ve decided to keep the sections separate but rename the “Background” to “Related work”. When including the “Related work” in the introduction, the section feels somewhat bloated to me. Closely linked, I refrain from describing the pros and cons of the different methods in the “Related work” section because there is not much literature on the topic. The main statements I can make about pros and cons are derived from my benchmark later on, which feels inappropriate for a background section. At the same time, this makes it difficult motivating PyOcto through the disadvantages of other methods. Instead, I opted for more broad statements in the introduction, a brief description of the algorithms in the “related work”, and lastly a more in-depth analysis in the results part of the manuscript.

L164: can you clarify what is meant by “greedy”?

> I’ve added a short explanation of the term “greedy algorithm” to the manuscript. (L165-167)

L175: Maybe change “PyOcto implements two velocity models” to “PyOcto can implement two types of velocity models”.

It’s a bit of a nuance but I decided to keep the formulation unchanged. The first formulation is meant to suggest that these are implemented, i.e., readily available. The second formulation suggests that no further velocity models could be implemented. While I’m not planning to do that for now, there is no principled reason not to do so.

L195: Maybe (re-)cite some of the DL-based associators being referred to.

I’ve added relevant citations here. (L205)

L180: Maybe I missed the context, but what is the symbol used here?

It’s Landau notation, i.e., asymptotic complexity of algorithms (https://en.wikipedia.org/wiki/Big_O_notation). Note that the typesetting in Latex differs a bit from the one on Wikipedia. For simplicity and to avoid confusion, I’ve exchanged the symbol with “constant time”. (L188)

L244 & L253: Is it worth stating the default values for association_cutoff_distance and min_pick_fraction?

I’ve added the default values to the text. (L266-267 / L275-276)

L283-284: Maybe add “(see Data and Code availability for details)”.

I’ve included this cross-reference now. (L305)

L405-407: in my opinion, I don’t think these two sentences need to be said. Feel free to delete it if you agree.

These sentences mostly serve as a warning. From my experience supporting issues for SeisBench, some users need to be made aware that simplistic demo workflows are not necessarily appropriate for application in research without further refinement.

L422-423: I feel that this comparison/fit with Omori decay needs to be quantified or at least shown in a Supplementary figure. Maybe the deviation from Omori is a real feature in the hours-days after a large earthquake?

I have exchanged the argument for a more simple justification of short term incompleteness: the number of events stays constant within the first 4 days, which is very unlikely in an aftershock sequence. Whiles it might indeed be interesting to study deviations from Omori’s law, this would be out of scope for this study. (L467-469)

L390 onwards: I may have missed it, but I didn’t say where the a priori homogenous and 1-D velocity models were defined or cited.

Thanks for pointing this out. The velocity model and hyperparameters have been added now. (L440-441)

L426: I’m not sure what “0.6 S picks” means. Is this a fraction? If so, is the word “times” missing? Maybe better to express it as a percentage?

It's referring the the number of picks, i.e., the number of S picks is on average 0.6 higher for REAL than PyOcto. As this is pointed out in the first part of the sentence, I decided to leave the formulation unchanged.

Figure 2: the red star isn't overly obvious. Could you please try a white or black outline on it?

I've added an outline and increased the marker size to improve visibility.

Figure 5: I was wondering if it is helpful to include the stations used on one of these maps?

I've added a supplementary figure showing the station configuration (Figure S1). I avoided adding the stations on one of the plots in Figure 5 to avoid visual bias between the different experiments.

Reviewer A:

This paper presents a new seismic phase association method, which is the task of grouping seismic picks (or phases) across stations into individual earthquakes. This is a fundamental task in building earthquake catalogs and is often the most computationally expensive and challenging step of an earthquake detection pipeline. This problem has received significant attention in recent years, since deep learning models provide high pick rates which increases the challenge of associating phases and identifying individual events.

PyOcto is based on using an OctTree-like partitioning in space and time to rapidly narrow in on promising candidate source regions. The paper shows this technique is very computationally efficient compared to several alternative (and recently proposed) associators, and is generally at least as accurate, if not more accurate, than the others. The paper is well written and the method is clearly explained. Synthetic tests of both shallow (crustal) seismicity, and subduction zone settings reveal various insights into the strengths/weaknesses of PyOcto and the other associators tested (GaMMA and REAL). A real test case on the 2014 Iquique earthquake sequence also shows a promising finding of many well located and detected events before and during the aftershock sequence.

Overall this is a strong manuscript, and the algorithm presented appears promising. The paper is clear enough to give readers insight into how the method works and may even inspire readers to make new algorithms for handling the association problem, as the paper illuminates several of the nuances of this challenge. I have several minor comments on some aspects of the synthetic data that I feel are too "simple", which the author might consider changing, or adding an additional synthetic test to test the models under these conditions. I have a few other minor comments on clarification. I expand on these below.

I thank the reviewer for their positive assessment of our work and for highlighting its context and relevance. I'm grateful for the collection of thoughtful remarks provided by the reviewer. Among their comments, the reviewer has proposed several extensions to the benchmark and to PyOcto. While without question interesting question were raised, I refrained from implementing several of these. Regarding the benchmark, I perceive this study as a brief analysis of the different associators rather than a fully comprehensive benchmark. Such a benchmark, would need to take substantially more questions into account, for example, the tuning of the model parameters. Regarding PyOcto, in my experience the simplistic assumptions, for example, on uncertainties or topography corrections, perform rather well in a wide range of cases. These assumptions are also commonly used in other association algorithms. However, I've added an explanation on extending PyOcto, highlighting that if required these extensions could be added in the future. Please find a point by point response below.

Comments:

1). In the uncertainty term in equation (1), it is not stated how this value should be set in general. Most importantly, it is not stated whether this is a fixed tolerance for all picks, or a tolerance that is proportional to the travel time. Most naturally, it seems it should be chosen per source-receiver pair as some fraction of the travel time (e.g., 1 – 5%). I would clarify this point.

PyOcto uses a constant uncertainty that is set in the velocity model. This is now clarified in the manuscript. While I agree that there are different ways to set uncertainties (absolute, relative to travel time, individual for each pick, …), for now I opted for the easiest option. This is similar to other associators, e.g., REAL and GaMMA, and gives good performance. However, I've also added an explanation on extending PyOcto, highlighting that the modular structure allows to quickly implement such extensions if required. (L109 & L306-307)

2). In lines 186 – 189, you refer to "local extrema". It is not clear to me what is meant by local extrema in this context.

I've added an explanation for the term and clarified that I mean local minima/maxima along the depth axis. (L195-197)

3). Line 190 – Using vertical incidence assumption for the elevation correction will indeed be quite inaccurate for the station elevation correction. Most eikonal solvers can provide incidence angles. Accurate station corrections can easily be obtained using trigonometric relationships and the true incidence angle at the reference surface elevation. This might be worth correcting if it is an easy fix, as this might introduce an unnecessary level of error (~1 s error for high elevation stations and grazing incidence angles).

Thank you for pointing this out. I checked and the eikonal solver I used does not return incidence angles. I'm also not sure how easy the correction term would be, as a grazing incidence angle would also change the pierce point of the zero-surface. For the scenarios I tested so far, the correction were sufficiently accurate. An alternative option would also be to run the eikonal solver independently for each station, allowing to incorporate the station height correctly upfront. For now, I refrain from these options, but I'd like to point at the option to extend PyOcto with such functionality if required in the future. (L305-307)

4). In line 215, you describe including all picks in a buffer window before each disjoint segment. It would help to explain why this is done, since it seems more natural to include picks in a buffer after each disjoint segment (to allow an origin anywhere inside the volume to have access to any picks that are produced by it, up until the max moveout time?).

> Thank you for pointing this out. The description in the manuscript was indeed incorrect (while the implementation was correct). I've updated the description to actually match the implementation. (L223-228)

5). In line 218, you say you de-duplicate the event catalogs. This is indeed very important. However, how is it done in practice? As I imagine some duplicates that occur might not be "exactly" the same origin time/location/set of picks, but only very close to within some tolerance. Have you accounted for this?

> Indeed, inaccurate origin times or locations might be detrimental to a deduplication. Therefore, the deduplication of PyOcto relies exclusively on the overlap between the sets of picks of two events and is tolerant to partial overlaps. I've added an explanation of the deduplication strategy to the manuscript. (L230-234)

6). For the optimizations of computational cost, in lines 226 – 227, you say that once a pick has been assigned to an individual event, it is no longer considered anymore. This seems like a greedy assignment, and it could have negative down-stream consequences? For instance, it would cause two "duplicate" events, if they did occur, to not have the exact same set of picks, and hence possibly be hard to identify. Also, what if the first event that's nucleated and established is actually false (and sub-optimal) compared to another true event, yet now the more optimal event has lost picks to this first source?

> As the reviewer correctly points out, this is a greedy assignment. However, a key point is that we traverse cells from many to few picks, i.e., a pick would always first be assigned to an event with many picks before it would get assigned to an event with few picks. Nonetheless, this might lead to incorrect assignments if events are extremely closely spaced. On the other hand, this optimisation is absolutely indispensable. If the picks would not be remove, they'd create many duplicate detections for the same event around, leading to an explosion in run time.

7). In lines 231 – 234, you describe the caching strategy of avoiding groups of picks that you have not been able to locate. However, what is the criteria for determining a "non-locatable" event? This doesn't seem to be explained anywhere, and is non-trivial, as there is no hard-and-fast rule for what determines a "non-locatable" event based on the picks. There is of course degrees of misfit/fit, and it seems a more explicit rule for determining this and a per-application tolerance level would have to be determined.

> An event is non-locatable if the determined origin does not correspond to sufficiently many picks. This condition depends on the tolerance for matching picks to events and the required number of picks. I've added an additional explanation here to clarify this. (L248-250)

8). In lines 238 – 248 you describe the strategy for avoiding spurious alignments of picks on distant stations. This is of course a very serious problem when it comes to associators based on only matching travel time alignment of picks. This approach seems sensible, but you might also consider remarking in the discussion that one of the main (conjectured) advantages of deep learning associators is that they can more naturally find the patterns of stations that make sense for a given event. It is primarily this issue that sets deep learning associators apart from non-deep learning associators.

> Thanks for pointing this out. I've added this (conjectured) advantage of deep learning associators to the background section now. (L90-93)

9). In line 301, you state that the random error added to travel times is drawn from a Gaussian with standard deviation 1% the travel time. This is actually a very small level of error and is not very realistic. I would recommend using 3%-5% for a more realistic test case (note that, it is not uncommon on reasonably large regional scales to have up to 10% error for some source-receiver paths). It in fact would be very interesting to see the results for both the 1% and 3 (or 5%) case. Perhaps it could be added as an additional supplemental figure.

> The Gaussian error used has a standard deviation of 1 % of the travel time, **but at least 0.4 s**. This means that the error condition of 1 % travel time only plays a role for travel times above 40 s. At these distance ranges, the 1D approximation should usually be good enough for such low errors. Two further observations justify this selection. First, PyOcto supports station residuals, i.e., errors from near-station effects can easily be mitigated. Second, systematic errors from inaccurate velocity models are correlated between stations with a similar azimuth from the event. This means that even for higher errors, events might be mislocated but will usually not be missed. However, there are other scenarios where larger deviations are clearly relevant, e.g., for volcanos or other regions with highly heterogeneous structure. I'd leave these experiments to future, more comprehensive benchmarks. To emphasize the the standard deviation of 0.4 s is the more relevant condition, I've reformulated the statement in the manuscript. (L325-326)

10). In the section of synthetic data generation (lines 295 – 305), you describe using the magnitude of each event to guide the probability of which stations observe a pick. However, it's not clear how this is done? More importantly, though, I am curious how many missing picks various stations have in general. One of the hardest parts of association is that some events will only produce arrivals on ~several stations, and some will produce arrivals on ~hundreds, yet the associator must handle both cases. Based on the current writing/tests, this aspect of the problem is not clearly addressed. In general, it would be good to test the models ability to detect events that are only recorded on very few stations and events recorded on many stations.

> This was indeed not specified in the original manuscript. I've now added a supplementary text providing the details on how the pick probabilities were determined. In addition, I'd like to refer to the implementation of the benchmark that is publicly

available. Regarding the range of picks, the benchmark covers a wide range: from 10 picks up too almost 200 picks in the shallow seismicity scenario. (Text S1)

11). In line 311, it is stated that at least 10 picks are required to declare an event detection. This is a fairly high number, and in practice will lead to many lost events. In general, the number should ideally be ~5 – 7. It might be worth running an additional test where you lower this number and analyze the associator performances (perhaps as an additional supplemental figure).

The number of picks to require for an event is a difficult questions. Personally, I do not perceive 10 picks as a particularly conservative threshold. Deep learning pickers are highly sensitive but at the same time, will also produce false detections. Unfortunately, they also have a tendency to create correlated false P and S detections. Nonetheless, from my personal experience, the optimal strategy for applying these models is to use a very low picking threshold, such as 0.05, i.e., risking high numbers of false picks, and then running an association step afterwards with a higher requirement in terms of the number of picks. Regarding the total number of picks, 5 is definitely a lower bound, given that 4 picks are already required to locate an event. From my experience with larger deployments, even requiring 7 picks often leads to false positives. Nonetheless, I agree that this question deserves further study. As suggested in my introduction of this response, I believe that this is out of the scope for this manuscript and should rather happen within a larger, more comprehensive benchmark.

Please see my response to question 16 too, where I've analysed the Iquique data while requiring fewer picks.

12). As a side note, by ensuring that all events in the synthetic catalogs have at least 10 picks (as stated in line 313), you are only trying to detect events that are "fairly easy", since there are so many picks available for each. A more realistic and challenging test would be to try and detect events with as few as five picks.

As mentioned above, the question of the number of picks is non-trivial. I personally perceive the case of the benchmark as realistic and would like to point out that the benchmark rather targets scenarios with very high seismicity rates than with very small seismicity. In the latter case, I agree that a different threshold might be appropriate.

13). In line 334, and elsewhere, it is made clear that GaMMA has run-time issues and does not converge. However I was under the impression that GaMMA uses DBSCAN to break up picks into groups before applying the actual EM algorithm to determine the association solution. Are you sure that you have run GaMMA with both DBSCAN + GaMMA, as it is used in practice? If not, that explains why it was never able to converge for large numbers of events, as the GaMMA algorithm really does need DBSCAN first (for better or worse). If DBSCAN wasn't used in these tests, I am not sure it's accurate to claim that it wasn't able to converge, since then it is not being used in the way intended.

I run GaMMA with both DBSCAN and the core GaMMA algorithm. However, at large number of picks, the algorithm fails to break up the picks into sufficiently small clusters.

Larger clusters take substantially longer to associate or don't converge at all. I've added this observation to the manuscript. (L421-L425)

14). I believe GaMMA does have an implementation of 1D velocity models online now. The author might consider using this version of GaMMA with 1D, for a more consistent comparison of the associators.

Thanks for pointing this out. I've now added GaMMA with a 1D model to the benchmark. For the Iquique sequence, GaMMA with a 1D model failed to converge for about 30 % of the days. Therefore, I have not added these results to the main text. Instead, the results are available in the supplement. (No line numbers as this concerns several changes)

15). In line 376, is it clear to you why GaMMA is performing so poorly in the subduction zone setting at >100 events? It also is somewhat confusing, since in the real application on Iquique, the performance of GaMMA seems reasonable and more comparable to the others than this result indicates (other than the possibly spurious shallow events far to the east).

I've added an explanation how the optimisation procedure of GaMMA likely leads to this result. The results from the Iquique sequence are actually consistent with the most similar scenarios in the benchmark (subduction, 500 events, 0.3 to 1.0 noise). Smaller differences could, e.g., be related to the different distribution of seismicity in the aftershock sequence compared to the longer term catalog. (L403-407)

16). Similar to an earlier comment, I am curious how the real data catalogs would appear if the minimum number of required picks was < 10 (as stated on line 404). Would there be many false and scattered events? This could be an interesting supplemental figure to include.

I've added an experiment with a reduced number of required picks (7 picks per event). The experiment is documented in the main text and in three associated supplementary figures. While the overall number of events grows with the lower threshold, the scatter of seismicity also grows substantially. It is unclear to what degree this is related to false detections and to which degrees to less accurate locations. However, from other experiments where I've relocated the events using NonLinLoc, I strongly suspect that many of the scattered events are false detections. This is part of my reasoning for choosing the higher pick threshold at 10 picks per event. (L505-512)

18). Line 418, it is stated that the catalogs developed with homogenous velocity models are slightly larger than 1D velocity models. This is a somewhat surprising finding, isn't it? Does it suggest that more false (or maybe split) events are being created with the homogenous velocity models?

This observation is most likely related to the different tolerance values for travel time residuals. The homogeneous models use a higher tolerance to account for the less accurate travel time calculation. This might lead to more false associations, but it's hard to identify these automatically. I've added a short remark to the manuscript. (L472-476)

19). The catalogs obtained in Fig. 5 are fairly impressive in terms of having little scatter and accurately resolved depths.

I thoroughly enjoyed reading this manuscript, and I think readers can learn a lot about the earthquake detection problem from this paper.

Thank you once again for the positive feedback and the thoughtful recommendations.

Reviewer B - Sacha Lapins:

This manuscript presents a new seismic phase association method based on efficient octree partitioning of the space/time search space used to identify and associate true seismic phase detections with a common source/origin. This approach and other algorithmic choices made by the author (e.g., use of EDT loss) seem to be inspired by the very popular and extremely well-tested NonLinLoc earthquake location package, often chosen for its robustness in the face of outliers and its computational efficiency. The proposed method appears to benefit from the same robustness and efficiency, which is clearly and fairly demonstrated by the author using synthetic and real-world case studies. Method limitations and parameter choices are also fairly presented and well explained. As such, this will likely be an incredibly useful publication and tool for the wider seismological community. Given its timely contribution, I am happy to recommend this manuscript for publication with only minor comments/edits (mostly typos).

I thank the reviewer for his positive assessment. Indeed, NonLinLoc was a useful inspiration, even though the idea of using space-time partitioning occurred to me in a completely different context. Thank you also for the recommendations regarding typos and formulation. I've implemented these. For brevity, I omit them from the response letter.

Minor comments / suggestions:

1. Line 218 – I may have missed it, but how are duplicated events identified and removed? Are there ever occasions where duplicated events might only have subsets of picks in common and, therefore, slightly different origins/locations? Or do they always contain the exact same picks (in which case I can see they would be easy to identify and remove)?

Please see the answer to point 5 of reviewer A.

2. Lines 425-432 / Figure 6 – You mention that REAL finds more S picks per event than PyOcto. To my eye, REAL also appears to consistently associate more P picks. Other than pointing out the higher number of S picks, you don't really mention why this might be in terms of the algorithmic choices between PyOcto and REAL, or relate this observation to your findings from the synthetic cases. Are REAL's additional picks likely to be errors? Or is PyOcto perhaps more conservative? I felt like there was a little bit more to unpack here.

Regarding the different number of picks, this is most likely related to the different tolerance criteria for matching picks to origins. While I've tried using consistent values across the different associators, the available criteria are defined slightly differently. I've added a paragraph discussing this. Regarding the higher number of P picks, I didn’t perceive this as a consistent feature, so I was not comfortable pointing it out. (L472-476)