First and foremost, we want to thank all the reviewers for their thorough reviews. We are especially happy about the high scores we have received from *Reviewer #3* and *Reviewer #4*, but also appreciate the in-depth analysis by *Reviewer #1* and the suggestions of *Reviewer #2*.

The problem we (and the NPU) are solving is to learn a subset of arithmetic operations (AOs), namely ($\times$, $\div$, x^w). Prior art either has problems with small and negative numbers (NALU) or implements only a subset of AOs (NMU). We have demonstrated these limitations in experiments 2 & 3. *Reviewer #2* was concerned about sufficient novelty because we (just) perform the algebraic operations in *complex* rather than *real* space. We believe that it is precisely this simplicity, which makes our approach convincing, because it does not require additional, more complicated mechanisms to work (like e.g. iNALU), and it fixes the shortcomings of prior work e.g. by a mathematically correct treatment of negative inputs. *Reviewer #3* is correct that relevance gates that are not exactly zero or one lead to a complex treatment of real inputs, which introduces an error in the result. We believe this error is actually an advantage because the optimization has to push the gate to either zero or one to remove the error. However, we agree that gates should converge to exactly zero or one, and we plan to address this in future work (e.g. by an additional penalization).
In the following three paragraphs, we address the criticism expressed concerning our experiments as they appear in the paper.

Experiment 1 - Equation Discovery Since the NALU reported problems with convergence, this experiment was set to demonstrate that the NPU can learn models of sequential data, which is notoriously difficult due to vanishing/exploding gradients, and bifurcations (also faced e.g. by RNNs). Additionally, small changes in the parameters can lead to substantial changes in the output after a few time steps. The experiment shows that the NPU can be used for sequential tasks, which we believe is owed to the combination of complex weights and the relevance gate.
Our second intention with this experiment was to outline how to utilize the transparency of the NPU as one part within a broader equation discovery framework. Since the NPU can represent more AOs than prior art, it is possible to fit the same data with simpler models (in terms of the number of parameters). This improves *transparency by decomposability* (see Lipton), which enables to extract a *practically useful* equation containing only a few terms. In the paper, we called this *interpretability*, which we will correct as suggested by *Reviewer #1*. We did not claim to recover the correct fSIR model, but rather something close to the original model. Finalization of the equation discovery would require more post-processing such as a detailed search of models near the obtained result, e.g., by using methods from binary neural networks or the regularization by penalizing entropy, as used in the GNN explainer. Considering the feedback on this experiment, we should have been more explicit about its purpose. We propose to change the title to "*A Step Towards Equation Discovery*", and to clarify in the text that the experiment is a proof of concept.
In response to *Reviewer #3*: The large MSE that some NPU realizations exhibit in Fig. 4 is caused by diverging models at the beginning of training (due to the difficulties named above), from which the models sometimes do not recover. We used ADAM to obtain a good initial solution further refined by LBFGS, which can improve MSE by around an order of magnitude.

Experiment 2 - Simple Arithmetic In this experiment, we wanted to show that the NPU can learn fractional power functions, and additionally, that it finds a solution where the NALU fails. While the NALU can learn $+$, $\times$, and $\sqrt{\cdot}$ in isolation (see Trask et al.), it does not converge for the more complicated task of learning those AOs in one model, as in our setup of this task. Schlör et al. suggested that this might be due to the gating between addition and multiplication paths in the NALU. We only tested one range in this experiment, because $\div$ and $\sqrt{\cdot}$ quickly become approximately linear away from zero, making the gradient signal to the correct solution (i.e., equation) weak. Training models on such a weak gradient signal is very sensitive to step-size in SGD and requires increasing the batch sizes. We have therefore considered training on a range with stronger gradient signals to be a fairer comparison between models.

Experiment 3 - Large Scale Arithmetic In this experiment, we aimed to stay close to the established benchmark by Madsen & Johansen. We have slightly modified it since the original setup of the task makes it difficult to learn $\div$, which is again due to weak gradient signals at the effectively sampled ranges. Specifically, their experiment applies a single AO to the sum of two overlapping subsets of a vector. For example, summing 50 numbers from a vector with samples from $\mathcal{U}(0,1)$ - via the central limit theorem - results in samples for the AO of interest from a narrow Gaussian centered at $\mu = 25$. For $+$ and $\times$, which were the interest of the NMU experiments, the signal is still strong, but for $\div$, this is results in a weak signal and thus a very hard task for optimization. Therefore, we chose an easier inverse x^{-1}, which also proves our point (that the NPU can represent and learn division by a variable).

Finally, we would like to thank the reviewers again for their effort and offer to update our description of experiment 3 as stated above, and to add a negative example to the broader impact section (the misuse of the setup in experiment 3 for real-world COVID-19 predictions) in case our paper should be accepted.