

– **Internship report** –
2022

Transmission loss estimation using Machine Learning

Edouard FORESTIER

Student at ENSTA Paris

Supervisors : Sven Peter NÄSHOLM, Quentin BRISSAUD

CEA contact : Alexis LE PICHON

Internship from 8 March 2022 to 30 July 2022

Acknowledgments

I want to thank NORSAR, the CEA of Bruyères-le-Châtel and my school, ENSTA Paris, for this great internship opportunity, which helped me to discover a lot of things and acquire new skills. A great thanks to my supervisors at NORSAR, Sven Peter NÄSHOLM and Quentin BRISSAUD who made sure that the project was going well, were available to help when needed, and made sure that it was an enjoyable experience. A special thanks to Alexis LE PICHON who helped to set up this internship, and followed the progress made as well as giving advice during the project. I also want to thank all the people I met at NORSAR, as it was a very nice work environment.

Funding

This internship was supported by the project *Middle Atmosphere Dynamics: Exploiting Infrasound Using a Multidisciplinary Approach at High Latitudes* (MADEIRA), funded by the Research Council of Norway basic research programme FRIPRO/FRINATEK under Contract No. 274377.

Contents

Acknowledgments	1
Funding	1
1 Introduction and motivations	3
2 A more realistic simulation dataset	3
2.1 Data collection and slice extraction	3
2.2 Atmospheric profiles extraction	4
2.3 Running infrasound simulations	5
3 Machine learning architectures	5
3.1 Input and output pre-processing	5
3.2 Improving the ML architecture	5
4 Training and evaluation of ML models	6
4.1 1000 km dataset from <i>Brissaud et al. (2022)</i>	7
4.2 4000 km dataset from Alexis Le Pichon	8
4.3 New 4000 km dataset	9
5 Comparisons with Alexis Le Pichon’s updated semi-analytical formula	11
6 Tests on the 2020 explosion in Beirut	11
7 Conclusion and perspectives	12
A Other ML architectures	14
A.1 Stacking more GRUs	14
A.2 Weights changing with distance to source	14
A.3 Transformers	14
A.4 Sequences return	14
B Prediction tool	14
C Error histograms	14

1 Introduction and motivations

Predicting the spatial amplitude distribution of low frequency sound, called infrasound (< 20 Hz), from surface sources is key to understanding observations at ground-based stations. For example, it is important to have a reliable quantification of the propagation path effects on the waves when characterizing infrasound sources like meteorites, earthquakes, nuclear and non-nuclear explosions, sea-generated microbaroms, or gas flares. However, infrasound propagation in the atmosphere is a complex problem owing to its frequency dependency and the time and spatial variability of atmospheric conditions (e.g., wind and temperatures).

To characterize the attenuation of infrasound amplitude with distance, researches typically use ground-to-ground infrasound Transmission Loss (TL) metrics in db which can be modeled using full-waveform simulations (Waxler *et al.*, 2021). However, full-waveform simulations are computationally expensive which prevents large-scale modeling or any real- or near-real-time applications. Currently, many research studies and operational signal analysis teams use semi-analytical models developed to provide first-order transmission loss predictions, up to a range of ~ 1000 km, as an alternative to time-consuming simulations (Le Pichon *et al.*, 2012). These approaches are simplified both in terms of what information about the atmosphere they use and in terms of the model buildup.

To further improve the accuracy of this model while keeping the computational cost low, NORSAR has developed a Machine Learning (ML) based model to recover transmission loss curves at regional distances with greater precision Brissaud *et al.* (2022). This model attempts to build an alternative mapping between a 2d wind model (height vs distance) and the ground TL. This allows for fast (< 0.05 s) and relatively accurate predictions (< 5 db on average) for the range of applications considered.

However, this 1000 km threshold is limiting for many applications at global scale such as microbarom modeling which generally requires > 2000 km propagation paths (Vorobeveva *et al.*, 2021). Additionally, in the first ML implementation, causality of the wave propagation process, i.e., TL at a given distance from the source should not be dependent of winds beyond that point, was not enforced owing to the use of a standard Convolutional Neural Network (CNN).

In this internship project, the ML approach has been further developed and evaluated using new ML architectures enforcing causality, a larger simulation dataset with more realistic wind models extended to larger propagation ranges (up to 4000 km), and improved semi-analytical models. These new architectures are based on recurrent neural network (RNN) tools as well as convolutional neural networks (CNNs). Comparisons with real observations are also presented.

2 A more realistic simulation dataset

To build and improve on the previous ML NORSAR study, this new dataset must include 4000 km wind profiles/transmission loss with more realistic small-scale perturbations. In order to keep the simulation cost manageable for NORSAR’s computational infrastructure, the spatial step between each wind profile, extracted from the ERA5 reanalysis atmospheric model product dataset, should be greater than 1 degree. Additionally, these atmospheric slices should be extracted all around the earth, at different times of the day and years/seasons to cover most different weather configurations. Similarly to Brissaud *et al.* (2022), this simulation dataset should be extended by considering various gravity wave perturbations, upwind and downwind scenarios, and random source frequencies for NCPA simulations. Finally, to ensure a good predictive accuracy, the final dataset size should be in the typical ML range for medium-size networks $> 10^5$ simulations. This simulation number is given by the following relation:

$$N_{\text{simulations}} = N_{\text{slices}} \times N_{\text{perturbations}} \times N_{\text{frequencies}} \times N_{\text{upwind/downwind}}, \quad (1)$$

with N_{slices} the number of slices, $N_{\text{perturbations}}$ the number of random gravity wave realizations, $N_{\text{frequencies}}$ the number of random frequencies and $N_{\text{upwind/downwind}} = 2$. If the desired number of simulations is $N_{\text{simulations}} \geq 200000$, if $N_{\text{perturbations}} = 4$ and $N_{\text{frequencies}} = 5$ then $N_{\text{slices}} \geq 5000$.

2.1 Data collection and slice extraction

Data collection and slice extraction, i.e., projection of atmospheric parameters along a given slice from a start to an end point, are critical to capture the diversity of weather scenarios. However, these steps can be time consuming as (1) the bandwidth to download data from atmospheric model providers such as ECMWF’s ERA5 dataset is limited, and (2) creating slices involves numerous file reading and writing operations.

Data collection In the previous work *Brissaud et al.* (2022), slices were created from specific netCDF files, corresponding to rectangular regions around the slice, downloaded individually from Copernicus servers. As the slices are 4 times longer than in the previous work (from 1000 km to 4000 km), yields much heavier netCDF files which would saturate NORSAR’s available storage. Moreover, the download waiting time on Copernicus is dependent primarily on the number of requests and less on the size of the region requested.

In this work, we downloaded global netCDF files covering the whole earth, with 1 degree steps in longitude/latitude, at specific time. Thus, with a small number of requests, and therefore files (each file having a size of around 700 Mb), we can cover a large range of weather scenarios thanks to the variability between hemispheres and the day/night cycle. In the final dataset, we downloaded 6 files, at different years, months, hours, covering the whole earth, as is displayed in Table 1.

Table 1: Summary of the parameters used to download ERA5’s netCDF files to build the new 4000 km dataset.

number	year	month	day	hour
1	2011	January	03	00h00
2	2013	September	28	18h00
3	2014	May	19	06h00
4	2016	July	18	12h00
5	2018	November	10	15h00
6	2020	March	08	03h00

Slice extraction By uniformly sampling random start and end locations from these files converging various azimuths and locations (longitude / latitude coordinates for origin point), we created 5000 slices per file, extending from a distance of 0 km to 4000 km with 100 km steps. Slices might cover similar regions since we randomly sampled locations which would promote overfitting in our ML model. To remove slices that are too similar we considered a similarity criterion based on the overlap between regions around each slice. Each region’s boundary is defined by a given distance from the slice.

We then pruned slices from the dataset considering a threshold on the similarity criterion set here at a value of 33% overlap between two slices (overlap between slices A and B = intersection area value divided by the “area” of slice A, the area is defined by extending all directions around to slice by 1 degree). The value of 33% allows to cover various azimuths from the same start location. Using this slice removal procedure over our 5000 original slices leads to a final number of 4451 slices per file. Figure 1 shows the slice distribution where each color corresponds to a different netCDF file. Despite our pruning procedure, we still keep a uniform distribution of locations along latitudes and longitudes.

2.2 Atmospheric profiles extraction

ERA5: atmospheric specifications Once the distribution of slices is finalized, we must extract atmospheric parameters along each slice to run simulations and train our ML model. This task was originally parallelized where each CPU would extract profiles for a given set of slices, open each corresponding file, and generation the simulation folders. In this study, the main challenge with the atmospheric profiles extraction is to avoid loading the 0.7 Gb netCDF files (instead of 0.03 Gb in the previous study) on each CPU, which can lead to memory saturation. The code has therefore been modified to allow the loading of a only small areas of the netCDF file containing the slice instead of the entire file which drastically reduce the memory footprint. Major improvements have been made to the original Python codes to reduce the computational time. In particular, the reading phase has also been optimized to allow for better performances.

MSIS: Higher-altitude atmospheric specifications To properly account for thermospheric refractions, profiles must range up to 120 km altitude. Because ERA5 profiles are limited to 80 km altitude, we extend atmospheric specifications up to 120 km using the empirical model MSIS (*Picone et al.*, 2002). Similarly to *Brissaud et al.* (2022), we stitch ERA5 and MSIS models together by interpolating atmospheric parameters, over a defined transition zone ($80 \text{ km} \pm 5 \text{ km}$), using cubic splines.

Downwind and upwind scenarios In order to introduce more diversity in the dataset while keeping the computational cost low, we consider both upwind and downwind scenarios on each slice. The wind profile corresponding to the original 0 to 4000 km range corresponds to the downwind scenario while the one corresponding to the 4000 to 0 km range is the upwind scenario. This essentially flips the wind directions which can dramatically change the amount of infrasound energy refracted to the ground when wind ducts are present.

Adding small-scale perturbations ERA5 and MSIS models do not resolve small-scale perturbations owing to gravity-wave breaking in the atmosphere (*Gardner et al.*, 1993). Therefore, to produce more realistic atmospheric models and introduce more diversity in the dataset, we add gravity-wave perturbations to each model using the Gardner model (*Gardner et al.*, 1993). For each slice created after the upwind/downwind step, 4 different Gardner-like scenarios are computed (including a scenario without any perturbation). Once the 120 km wind profiles are computed using MSIS, Gardner-like perturbations are added. In the present study, we considered range-dependent gravity-wave perturbations as the domain extent is much larger than in the previous study. Perturbations are computed using a code from the CEA (France) translated into Python. Each scenario is created based on a NumPy seed to control the random process and the seed is stored with the slice information for reproductibility.

Introducing the frequency dependency The propagation of infrasonic energy is not only wind and temperature dependent but also frequency dependent. Therefore, we must introduce the frequency parameters when building the simulation dataset to cover a wide variety of source conditions. For each Gardner scenario, we uniformly sample five different frequencies between 0.1 and 3.2 Hz for to simulate the ground TL. They are stored with the slice information for reproductibility.

Issue with the current dataset and additional information An error occurred during the dataset generation process which was only caught later on. All profiles were determined using the MSIS model computed for November, 10 2018. Thus, the variability in the wind profiles may not be as good as expected. However, it should be noted that (1) spatial variations of MSIS models are included and provide a lot of variability, and (2) the added Gardner perturbations may compensate this lack of variability in these high altitude zones. Note also that for reproductibility, slices will be assigned an ID and some information about the origin slice, downwind/upwind scenario and then allows identification for the slice’s place among the Gardner/frequency scenarios.

2.3 Running infrasound simulations

Once the simulation are folders are created, we computed TLs using the ePape software from the NCPA suite (*Waxler et al.*, 2021). We used the latest and updated version of the ePape code which corrected absorbing boundary conditions from the last study. However, we considered the same simulations parameters (see *Brissaud et al.*, 2022, for more details).

3 Machine learning architectures

3.1 Input and output pre-processing

Before ML models are trained, input wind models and output TLs must be pre-processed to facilitate the training and evaluation processes.

Input encoding Inputs, i.e., wind model variations, must be encoded to provide a better distribution of values before injecting them into the first ML layer. The encoded inputs are 1-channel (black and white) images downsampled from the full 40x1000 wind profiles, which have 40 different ranges (from 0 to 3900 km with 100 km step, the 4000 km profile being excluded) and 1000 altitude points (from 0 to 120 km). The new images are 40×50 with 40 ranges and 50 altitude points.

Encoded outputs In *Brissaud et al.* (2022), the encoded outputs were originally a 500-point interpolation over each TL, i.e., resampled from the original TL to reduce the number of parameters and ensure consistency across all simulations (which can vary in TL data points). This value has been modified by (1) lowering the number of output data points down to 400 points to further reduce the number of parameters needed in ML model, and (2) smoothing the resulting TLs to facilitate convergence. The downgrading and smoothing procedures are acceptable from an accuracy standpoint since this tool attempts to provide reasonable solutions but does not attempt to capture the entire TL spectrum. Higher-frequency oscillations are generally dependent on finer-scale variations of the atmosphere which are poorly constrained and our model would therefore provide an average solution. The smoothing procedure is based on the ‘lowess’ routine from the library ‘statsmodels’. TLs are then scaled using a standard scaling, removing their mean and scaling them with their standard deviation. In Figure 2, we show comparisons between the original (red) and smoothed (blue) TL curves.

3.2 Improving the ML architecture

To build and improve on the CNN model used in the previous study (*Brissaud et al.*, 2022), we explored various ML architectures to ensure causality and improve accuracy.

2D CNN This architecture is the one presented in *Brissaud et al. (2022)*. It is based on a standard image-processing tool called CNN are used to meaningful multi-scale patterns in the wind maps that explains the ground TLs. The CNN output is a translation-equivariant response known as feature maps. CNN features, i.e., outputs of the 2D CNN layers, are then concatenated to the frequency and used as inputs for the fully-connected layers (each neuron in one layer is connected to all neurons in the next layer). The final output corresponds to the encoded TL.

1D CNN with GRU As a classic CNN architecture looks for patterns which are invariant with translation, it might produce non-causal relationships between winds and TLs with range. To enforce causality in the network architecture, we investigated Recurrent Neural Networks (RNNs, *Hochreiter and Schmidhuber (1997)*, *Cho et al. (2014)*). RNNs introduce causality by constraining the relationships along one dimension of the input. RNNs have traditionally been employed to capture time dependency for video or signal analyses where only the information up to a given timestep is used to infer an output at the same timestep. Note that considering range dependence instead of time dependence is essentially the same problem. RNNs are mostly based on two similar layers: LSTM (*Hochreiter and Schmidhuber (1997)*) and GRU units (*Cho et al. (2014)*). We used GRU units, as they involve less parameters and are generally easier to optimize over small datasets.

As we introduce more parameters than for the original 2D CNN approach, we replaced the 2D convolution layers with 1D distributed convolution layers, which extract features from all altitude layers at a given range (see Figure C). We then used the output of each 1D convolution as the input for our GRU units. GRU outputs are then connected to a fully-connected neural network that outputs the ground TL. The GRU unit helps computing an estimate of the TL at a given range using only the information up to this range. It takes into account the fact that the profile at a specific range step has an impact on the transmission loss at larger distances from the source but not at before.

However, as the frequency was first concatenated before the fully-connected layer and was not considered as an input for the GRU, we decided to test an alternative architecture as shown in Figure 5.

Despite this improvement in the capture of causality, the fully-connected layers may still shuffle the outputs of the RNN units. Another solution would be to replace the fully connected layers by an architecture like a WaveNet (*Oord et al. (2016)*), which is a neural network that takes into account causality with connections only in one direction (the increasing range direction). This idea is developed in Section A.1.

1D CNN with stacked GRUs To further improve model predictive accuracy, additional GRU layers can be added, i.e., stacked, to capture more complex causal relationships. This architecture has originally been used to improve the predictions of future time steps or next words in a sentence. However, this structure is not limited to predicting future data, as GRUs have proven to be able to be used for computing time series from same time range data (*Xia et al. (2021)*). By constraining the flow of information during training and inference only towards positive times or increasing ranges, it allows the model to converge in much fewer epochs. Here, the wind profiles have an effect on the TL for equal or at further distance from the source.

To further improve the first GRU implementation described in Figure C, we attempted to remove the final fully connected layers (see Figure C), which could shuffle the range-dependent vectors coming from the stacked GRUs and impact causality. Finally, the frequency has been moved to an earlier stage of the ML architecture, and concatenated to the features extracted through the 1D convolution layers, allowing the GRUs to take frequency dependency into account.

4 Training and evaluation of ML models

Training a ML model corresponds to the iterative optimization of the model parameters, known as neurons and defined by a set of scalar parameters, for a given objective, also known as misfit. Once a model is trained, computing errors and comparing accuracy against other models is key to understand a model’s weaknesses and assess the reliability of our ML architectures for a specific set of applications. The accuracy on the test dataset is highly dependent on the choice of dataset, its diversity in terms of input and output pairs and the quantity of data provided. In this section, different figures will be presented to assess and compare the different ML models over different datasets, namely, convergence curves, error distribution on the test dataset, comparisons between predicted and true TLs, and TL curves vs C_{eff} .

Convergence curves The model’s convergence curves give information about the final root-mean-square error (RMSE) (after scaling, this is not decibels), which is one of the main criteria to evaluate the performance of the model. The RMSE is calculated by comparing the ML prediction of the TL with the smoothed input of the training.

Error distribution The error distribution shows the variation of the error with the range, frequency and C_{eff} , which may help to decide which model should be used, as a model performing better on further ranges may for

example be more useful in certain situations. These errors are computed by comparing to the NCPA transmission loss and not the smoothed inputs of the training.

ML predictions vs true TLs Comparison figures between predicted and true outputs illustrates the ML model performances vs distance from the source for a given wind scenario. ML predictions will be shown in blue, along with the ML uncertainty in shaded blue color as well as the simulated NCPA transmission loss (curves called Parabolic Equation, or PE). Uncertainties were computed using the following process: In the test dataset, range-dependent errors $ERR(x)$ between ML predictions and NCPA simulations have been computed as $ERR(x) = |NCPA(x) - ML(x)|$. Then, quantiles of errors have been computed for each range, and for all the test dataset. For example, at 1500 km from the source and for a specific frequency range it is possible to know the 95% quantile, below which 95% of the error values for all the test dataset are. Additionally, wind profiles and C_{eff} ratio in function of range are shown to facilitate the interpretation of TL variations with distance.

Variations of TL with C_{eff} Plotting the variations of TL with C_{eff} for both ML and true predictions illustrates how much physics can be learned by our model. In particular, if the ML model can capture (1) overall TL trends with distance, and (2) acoustic shadow zones, regions where the infrasonic energy is not fully refracted back to the surface. This information will be shown for the new 4000 km dataset and the 1000 km one from *Brissaud et al.* (2022). This plot is an interpolation of the TL values on a C_{eff} and range grid, the frequencies chosen: C_{eff} is shown in the range $0.85 - 1.2$ and f_0 equal to 0.15 or 2 Hz. The interpolation is carried out on the following grid: 400 ranges between 2 and 3999 km, and 100 C_{eff} between 0.85 and 1.2, then the results are plotted. The results are shown for $[0 - 4000]$ km and $[0 - 1000]$ km for the new 4000 km dataset, and only in the 0 to 1000 km range for the other one.

4.1 1000 km dataset from *Brissaud et al.* (2022)

Dataset properties This dataset includes wind profiles generated by range-dependent 80 km ERA5 profiles extended to 120 km using MSIS, and on which range-independent Gardner perturbations have been added. It includes 41946 profiles/TL couples. The distribution of wind and TL profiles with distance and altitude is shown in Figure 7.

ML model properties The parameters for the models discussed are displayed in Tables 2 and 3. Table 2 highlights the properties of the fully-connected (FC) layers, and the activation function used, and Table 3 shows the 1D or 2D CNN layers properties (depending on the chosen architecture). The “size factor” is a factor applied for each following layer. For example, if the initial size is 50 and the size factor is k with a number of layers of 3, layer one will have an output size of 50, layer two an output size of $k \times 50$ and layer three $k^2 \times 50$. Please note that when the number of layers is 1, the size factor is useless.

Table 2: Parameters of the models (1)

type of model	learning rate	number of FC layers	FC initial size	FC layer size factor	activation function
2D CNN	0.0001	2	50	2	relu
1D CNN with GRU	0.001	1	50	-	relu
1D CNN with stacked GRUs (size 110)	0.00085	1	50	-	relu
1D CNN with stacked GRUs (size 400)	0.00035	1	50	-	relu

Table 3: Parameters of the models (2)

type of model	1D/2D filter number	filter initial size	filter size factor	GRU output size
2D CNN	3	2^5	2	-
1D CNN with GRU	1	2^5	-	250
1D CNN with stacked GRUs (size 110)	1	2^5	-	110
1D CNN with stacked GRUs (size 400)	1	2^5	-	400

2D CNN The approach CNN has been tested in *Brissaud et al.* (2022). This is a baseline for improving the results with the new architectures. The results shown here are slightly different as the size of the outputs changed from 500 to 400, but remain close to what was obtained previously according to Figure 8. It should be kept in mind

that the RMSE values in Figure 8 are not in decibels because they are computed after a data scaling, and more information is given in Section 4. The RMSE curves in Figure 9 give global information about the accuracy of the method in decibels, and especially its evolution with the range, frequency and C_{eff} . Subplot *a*) shows that the lower the frequency, the better the accuracy. Furthermore, subplot *b*) also shows the relationship between the error distribution and range (suggesting that at large ranges the accuracy is better than at close ranges), but the distribution keeps the same shape for different ranges, which is not the case in subplot *a*) for the frequency.

In Figure 10, the plots provide a visual comparison between the predicted and true TLs (the shown results are chosen randomly). It helps to check if the first bounces are well captured by the model for example. Here, this model provides accurate results for slices 1702487 and 1704095 but fails to give satisfactory results for slice 103220. The predicted TLs for range 0 – 200 km are usually far from the expected results (on average around 9 dB according to Figure 9), which will be improved in the next models used.

1D CNN with GRU

The results shared here, taking into account the introduction of a GRU unit and an optimization of the architecture (Bayesian research), are comparable with what is obtained in *Brissaud et al.* (2022) in terms of RMSE error. Comparing to what is shown in 4.2.2, the prediction is significantly better, especially for slices 310 and 1702487 where the first and second returns are better fitted.

1D CNN with stacked GRUs

Recall that this architecture couldn’t be optimized with the provided tools, as the training is much longer than the previous architectures. Thus, the results here are shown for 2 different choices of GRU output size (110 or 400, based on previous results) but are probably not the best possible.

Again, the MSE/RMSE errors are comparable to what we had previously. However, as this architecture proved to be particularly efficient on the other datasets, the lack of performance suggests that this dataset is not large enough to allow better fits.

GRU size 110

GRU size 400

Comparisons between the models

Table 4 shows a comparison summary between the final validation/training RMSE of the different models (after scaling, this is not decibels). As explained in Section 4, the RMSE values here are based on the smoothed TL curves and not the original ones.

Table 4: Comparisons between the models

type of model	validation RMSE	training RMSE	epochs
2D CNN	0.56	0.35	48
1D CNN with GRU	0.50	0.17	58
1D CNN with stacked GRUs (size 110)	0.50	0.21	30
1D CNN with stacked GRUs (size 400)	0.50	0.18	29

4.2 4000 km dataset from Alexis Le Pichon

Dataset properties This dataset is composed of range-independent wind profiles perturbed with range-dependent Gardner perturbations. This is a fairly small dataset (8000 profiles/TL pairs) compared to *Brissaud et al.* (2022) and the present study. Due to its simplicity this dataset does not exhibit much variability in terms of effective velocities and distribution of wind ducts with altitude (Figure 24). Therefore, ML algorithms are expected to show excellent accuracy but also to not generalize well to new unseen data outside of the narrow distribution of wind variations included in the dataset.

Model properties ML model parameters used in this testing round are displayed in Tables 5 and 6. Table 5 highlights the properties of the fully-connected (FC) layers, and the activation function used, and Table 6 shows the 1D or 2D CNN layers properties (depending on the chosen architecture). The “size factor” is a factor applied for each following layer. For example, if the initial size is 50 and the size factor is k with a number of layers of 3, layer one will have an output size of 50, layer two an output size of $k \times 50$ and layer three $k^2 \times 50$. Please note that when the number of layers is 1, the size factor is irrelevant.

Table 5: Parameters of the models (1)

type of model	learning rate	number of FC layers	FC initial size	FC layer size factor	activation function
2D CNN	0.000538998	2	43	1.9622	relu
1D CNN with GRU	0.001	1	50	-	relu
1D CNN with stacked GRUs (size 110)	0.00085	1	50	-	relu
1D CNN with stacked GRUs (size 400)	0.00035	1	50	-	relu

Table 6: Parameters of the models (2)

type of model	1D/2D filter number	filter initial size	filter size factor	GRU output size
2D CNN	1	2^4	-	-
1D CNN with GRU	1	2^5	-	250
1D CNN with stacked GRUs (size 110)	1	2^5	-	110
1D CNN with stacked GRUs (size 400)	1	2^5	-	400

2D CNN In Figure C training and validation curves show similar trends and errors, which indicates that the 2D CNN model does not exhibit any strong overfitting. Predictions fit fairly well with the true TLs (Figure 26): the first acoustic shadow zone is fairly well captured, and then the general trend of the predicted TL curve fits the test data.

1D CNN with GRU When combining distributed 1D CNNs with a GRU unit we observe a slight improvement in the final accuracy compared to the 2D CNN (Figure C). However, the training convergence decays much faster than the testing one indicating a potential overfitting for a large number of epochs. The first and second bounces seem to be accurately captured, showing an improvement compared to the 2D CNN in terms of error distribution (Figure 29). This suggests that the partial constrain of causality through the GRU unit might improve the overall convergence of the algorithm.

1D CNN with stacked GRUs - GRU size 110 Stacking GRU units still shows evidences of overfitting for large epochs (Figure 31) but leads to slightly better accuracy over the validation dataset (Figure 32). This overfitting problem might be due to an overparametrization of the model.

1D CNN with stacked GRUs - GRU size 400 Increasing the size of the stacked GRU units does not drastically change the convergence results when comparing Figures 31 and 34. Yet, this model leads to the best error distribution for this dataset (Figure 35), capturing both the first and second bounces, as well as the asymptotic behavior of the TL curve at large range from the source. It hints that stacking GRU units helps to better learn non-linear relationships between winds and TLs but a careful optimization of the architecture is required to avoid a strong overfitting.

Comparisons between the models Table 7 shows a comparison summary between the final validation/training RMSE of the different models (computed over normalized TLs, i.e., not in db). Note that, as explained in Section 4, RMSE values here are based on the smoothed TL curves and not the original ones.

Table 7: Comparisons between the models

type of model	validation RMSE	training RMSE	epochs
2D CNN	0.26	0.24	120
1D CNN with GRU	0.22	0.13	61
1D CNN with stacked GRUs (size 110)	0.22	0.14	50
1D CNN with stacked GRUs (size 400)	0.21	0.12	60

4.3 New 4000 km dataset

Dataset properties This dataset benefits from a larger diversity of input wind models and therefore TL outputs due to the significant increase in the dataset size compared to *Brissaud et al.* (2022) which should have a direct

impact on RMSE errors. Yet, by increasing the simulation range, we also increased the problem complexity which would require a large dataset to be accurately captured. The properties of the dataset are described in Section 2.

We can also compare the figures with the TL mapping as a function of the range and C_{eff} , checking that the dataset properties are indeed the same after this smoothing.

Model properties The parameters for the models presented in the previous sections are displayed in Tables 4.3 and 4.3. Table 4.3 highlights the properties of the fully-connected (FC) layers, and the activation function used, and Table 4.3 shows the 1D or 2D CNN layers properties (depending on the chosen architecture). The “size factor” is a factor applied for each following layer. For example, if the initial size is 50 and the size factor is k with a number of layers of 3, layer one will have an output size of 50, layer two an output size of $k \times 50$ and layer three $k^2 \times 50$. Please note that when the number of layers is 1, the size factor is useless.

Table 8: Parameters of the models (1)

type of model	learning rate	number of FC layers	FC initial size	FC layer size factor	activation function
2D CNN	0.001	2	49	1.7678	sigmoid
1D CNN with GRU	0.001	1	50	-	relu
1D CNN with stacked GRUs (size 110)	0.00085	1	50	-	relu
1D CNN with stacked GRUs (size 400)	0.00025	1	50	-	relu

Table 9: Parameters of the models (2)

type of model	1D/2D filter number	filter initial size	filter size factor	GRU output size
2D CNN	3	2^5	1.4317	-
1D CNN with GRU	1	2^5	-	250
1D CNN with stacked GRUs (size 110)	1	2^5	-	110
1D CNN with stacked GRUs (size 400)	1	2^5	-	400

2D CNN The 2D CNN architecture shows mixed results with a number of wind models in the validation set showing large errors.

1D CNN with GRU Similarly to what was obtained on the previous datasets, adding a GRU unit helps better capturing Tls over the validation set. In particular, a significant improvement can be observed for slices 223 and 610 (Figure 50), although some strong errors are still present. Interestingly, the second bounces are generally more accurately captured with this model.

1D CNN with stacked GRUs - GRU size 400 This model does not provide significant improvements to the previous one (Figure 48), but the MSE/RMSE error is still lower over the validation set (Figure 47).

1D CNN with stacked GRUs - GRU size 110

Again, a slight improvement for the MSE/RMSE error.

Comparisons between the models Table 10 shows a comparison summary between the final validation/training RMSE of the different models (RMSE are shown as normalized results, i.e., not in db). Note that, as explained in Section 4, the RMSE values here are based on the smoothed TL curves and not the original ones.

Table 10: Comparisons between the models

type of model	validation RMSE	training RMSE	epochs
2D CNN	0.40	0.19	34
1D CNN with GRU	0.39	0.13	57
1D CNN with stacked GRUs (size 400)	0.37	0.19	17
1D CNN with stacked GRUs (size 110)	0.39	0.27	8

5 Comparisons with Alexis Le Pichon’s updated semi-analytical formula

In *Le Pichon et al.* (2012), a semi-analytical formula, referred here as LP12, has been proposed to relate stratospheric winds, frequency and ground TL. The parameters of this regression equation were optimized using a small TL dataset generated using NCPA simulations and range-independent idealized atmospheric models. LP12 captures the main TL trends with distance and first and second acoustic shadow zones when strong stratospheric wind ducts are present but show large errors for upwind conditions or when tropospheric or thermospheric ducts are present *Brissaud et al.* (2022).

To assess if the updated attenuation law capture the physics behind infrasound propagation and provide meaningful comparisons with ML predictions, we plot TLs in function of C_{eff} and range as done in *Brissaud et al.* (2022). This map corresponds to an interpolation of the TL values on a $C_{\text{eff}} \times \text{range}$ grid, in a given frequency range. The parameters used in LP12 are the source frequency f_0 and the maximum effective sound speed ratio in the range 30–70 km C_{eff} . To produce LP12’s maps we ran 1000 computations using randomly sampled C_{eff} over a uniform distribution between 0.85 and 1.2 and f_0 equal to 0.15 or 2 Hz. The interpolation is performed on the following grid: 400 spatial steps between 2 and 3999 km, and 100 C_{eff} between 0.85 and 1.2. Maps are shown in Figure 55 for both dataset, i.e., [0-4000 km] and [0-1000 km].

Note that, building on LP12, CEA’s Alexis Le Pichon has worked on an updated version of the regression equations accounting for range-dependent variations and further ranges from the source. We compared our results to this yet unpublished new version of the attenuation law. As this attenuation law is likely to be further developed, quantitative comparisons are not shown here.

6 Tests on the 2020 explosion in Beirut

Our ML model can quickly generate TL estimates over large distances from a surface source. This can help localizing source location by computing the relative error between recorded signal amplitudes. Here, we attempted to locate the 2020 Beirut explosion using global infrasound recordings. To locate the source, we calculated the ground TL over 4000 km from a grid of possible source locations which were compared to the signals received at the stations I48TN and I26DE

Methodology We created 1000 slices, defined by a start and an end point, crossing each station location with different ranges and bearings. We then extracted wind profiles along these slices using the ERA5 file for August, 4th 2020, at 15:00:00 UTC (Figure C). Here we did not consider any gravity-wave perturbation.

Then, knowing the amplitude and frequency of the detected infrasound signals at the stations, it is possible to invert a prediction of the TL, and to predict the amplitude of an infrasound source for every slice. According to *Pilger et al.* (2021) and *Popov et al.* (2022), the received signals at I48TN and I26DE had the following properties: The

Table 11: Profiles characteristics

Station	Maximum peak-to-peak amplitude (Pa)	Dominant period at maximum amplitude (s)
I48TN	0.48	4.6
I26DE	0.12	4.6

TL map has to be computed for a frequency of $f = \frac{T}{2\pi} = 0.73$ Hz. As the TL calculated by the machine learning model is equal to $TL_{\text{dB}}(x) = 20 \log_{10}(\frac{A(x)}{A(x=1 \text{ km})})$, A being the amplitude of the infrasound signal, it is possible to retrieve the true TL with $A(x) = A(x=1 \text{ km}) \times 10^{\frac{TL_{\text{dB}}(x)}{20}}$. Knowing $A(x)$ and $TL(x)$ it is then possible to retrieve $A(x=1 \text{ km})$, the potential source, by inverting the relation. The amplitude at the station is derived from Table 11. This process has been carried out at several points and an interpolation has been made in Figure 57 for both stations.

Results The chosen model is the 4000 km one presented in Section A.1, which led to the best results in terms of validation.

Then it is possible to compute the difference in the interest zone (around Beirut). As seen in Figure 58, the misfit is the lowest South East of both stations, which corresponds to the location of Beirut. However, the global minima, deep blue line in Figure 58, are strongly shifted towards the West of Beirut and not accurately capturing the right location with a misfit larger than 10 Pa. This suggests that ML prediction errors and uncertainties in wind models

greatly impact the accuracy of this simple method. Our model tends to overpredict amplitudes at the stations and predictions are not seem to be consistent in terms of order of magnitude with the literature *Pilger et al. (2021)*. Note that the models trained on Alexis LE PICHON’s dataset gave also very similar results.

This inversion method could help constraining source location and amplitude if we reduce the overfitting of our ML model. Additionally, abberations in the output of the amplitude-based inversion model could be somewhat alleviated by including arrival times in the misfits as well as additional stations. Further research is needed to understand why the results do not match the observations well, but the tools have proven to be easy to use and fast to compute these source maps.

7 Conclusion and perspectives

This project built on a recent ML approach *Brissaud et al. (2022)* using more sophisticated ML techniques and a larger dataset. The new dataset takes into account more realistic wind perturbations and has an extended range in comparison to what has been done in *Brissaud et al. (2022)*, and the machine learning models trained on it show very promising results. Moreover, using a larger dataset implied optimising the code and the storage of the data, making the process faster and more efficient for future use. The new architecture that uses recurrent neural network tools provides significant improvements as shown in Table 12. The main results obtained during this project are displayed in Table 12, which sums up the comparisons between the different models and datasets.

Training results indicate that our models might overfit the training data. This could be explained by a few factors: (1) the distribution of TL curves is concentrated at values over -100 db which makes learning upwind simulations difficult, (2) the encoding of wind models might not be optimal. In particular, the downsampling step during pre-processing might lead to the loss of small waveguides important to explain surface TLs, (3) additional physics-based constraints are needed in the misfit to remove extreme predicted TLs and help convergence.

Table 12: Comparisons between all the models (RMSE values are not in decibels, and computed after data scaling)

dataset	type of model	validation RMSE	training RMSE	epochs
1000 km dataset from <i>Brissaud et al. (2022)</i>	2D CNN	0.56	0.35	48
	1D CNN with GRU	0.50	0.17	58
	1D CNN with stacked GRUs (size 110)	0.50	0.21	30
	1D CNN with stacked GRUs (size 400)	0.50	0.18	29
1000 km dataset from Alexis Le Pichon	2D CNN	0.26	0.24	120
	1D CNN with GRU	0.22	0.13	61
	1D CNN with stacked GRUs (size 110)	0.22	0.14	50
	1D CNN with stacked GRUs (size 400)	0.21	0.12	60
New 4000 km dataset	2D CNN	0.40	0.19	34
	1D CNN with GRU	0.39	0.13	57
	1D CNN with stacked GRUs (size 400)	0.37	0.19	17
	1D CNN with stacked GRUs (size 110)	0.39	0.27	8

This ML model allows for fast global scale simulation of infrasound propagation which paves the way for near-real-time monitoring of surface sources. We performed a first attempt to use the ML model to retrieve the location of the Beirut explosion using only amplitude data. Location accuracy is poor although the lowest values of the misfit points towards a large region around the location of Beirut. Improved predictions would depend on several key elements, namely, better ML prediction accuracy, increased number of stations, and more realistic input wind models. This tool might also offers a more reliable predictive model to study long-range microbarom propagation and study atmospheric dynamics (*Vorobeve et al., 2021*). Finally, the release of the Python code and associated dataset will facilitate the development of ML-aided monitoring approaches from the infrasound community.

References

- Brissaud, Q., S. P. Näsholm, A. Turquet, and A. Le Pichon (2022), Predicting infrasound transmission loss using deep learning, *Geophysical Journal International*, doi:10.1093/gji/ggac307.
- Cho, K., B. Van Merriënboer, D. Bahdanau, and Y. Bengio (2014), On the Properties of Neural Machine Translation: Encoder-Decoder Approaches, *arXiv*, doi:10.48550/arXiv.1409.1259.
- Gardner, C. S., C. A. Hostetler, and S. J. Franke (1993), Gravity wave models for the horizontal wave number spectra of atmospheric velocity and density fluctuations, *Journal of Geophysical Research: Atmospheres*, *98*(D1), 1035–1049, doi:10.1029/92JD02051.
- Hochreiter, S., and J. Schmidhuber (1997), Long short-term memory, *Neural Computation*, doi:10.1162/neco.1997.9.8.1735.
- Le Pichon, A., L. Ceranna, and J. Vergoz (2012), Incorporating numerical modeling into estimates of the detection capability of the IMS infrasound network, *Journal of Geophysical Research: Atmospheres*, *117*(D5), doi:10.1029/2011JD016670.
- Oord, A. v. d., S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu (2016), Wavenet: A generative model for raw audio, *arXiv*, doi:10.48550/arXiv.1609.03499.
- Picone, J., A. Hedin, D. P. Drob, and A. Aikin (2002), NRLMSISE-00 empirical model of the atmosphere: Statistical comparisons and scientific issues, *Journal of Geophysical Research: Space Physics*, *107*(A12), SIA-15, doi:10.1029/2002JA009430.
- Pilger, C., P. Gaebler, P. Hupe, A. C. Kalia, F. M. Schneider, A. Steinberg, H. Sudhaus, and L. Ceranna (2021), Yield estimation of the 2020 beirut explosion using open access waveform and remote sensing data, *Scientific Reports*, *11*(14144), doi:https://doi.org/10.1038/s41598-021-93690-y.
- Popov, O. E., A. A. Mishenin, S. N. Kulichkov, I. P. Chunchuzov, V. G. Perepelkin, and L. Ceranna (2022), The effect of the stratopause on the structure of the infrasound signal from the august 4, 2020, beirut explosion, *The Journal of the Acoustical Society of America*, *152*(99), doi:https://doi.org/10.1121/10.0012191.
- Vorobeva, E., M. D. Carlo, A. L. Pichon, P. J. Espy, and S. P. Näsholm (2021), Benchmarking microbarom radiation and propagation model against infrasound recordings: a vespagram-based approach, *Annales Geophysicae*, *39*, 515–531, doi:10.5194/angeo-39-515-2021.
- Waxler, R., C. Hetzer, J. Assink, and D. Velea (2021), chetzer-ncpa/ncpaprop-release: NCPAprop v2.1.0, doi:10.5281/zenodo.5562713, last accessed on 29 October 2021.
- Xia, M., H. Shao, X. Ma, and C. W. de Silva (2021), A stacked gru-rnn-based approach for predicting renewable energy and electricity load for smart grid operation, *IEEE Transactions on Industrial Informatics*, *17*(10), 7050–7059, doi:10.1109/TII.2021.3056867.

A Other ML architectures

A.1 Stacking more GRUs

Stacking more GRU units has been tested for both the new 4000 km dataset and the 1000 km dataset. Here, the tests were with 6 GRU units instead of 3.

It led to promising results, especially in the 1000 km dataset in which some TLs were still not well approached. This architecture brought very interesting improvements. In Figure 60, a comparison of what we had before and what was predicted with the 6 GRU units is shown.

It is possible to make the same kind of comparisons for the new 4000 km dataset, as in Figure 61.

1D CNN with GRU and causal neural network The idea was to change the fully connected network into something similar to a WaveNet (*Oord et al. (2016)*), which is a neural network with causal links (information can only go in one direction - the increasing range one). To do so, it is possible to use stacked 1D CNN layers with causal padding instead of the fully connected layers (Figure 62), thus creating the desired structure.

Some comparisons are shown in Figure 63.

A.2 Weights changing with distance to source

In order to make the algorithm focus on predicting better the first 1000 km of the TL, a custom loss has been created. It gives more weight to that part. However, the tests were not conclusive and could not be further explored in the given time, even if a small improvement has been highlighted.

A.3 Transformers

Transformers have also started to be explored, but for the same reasons, satisfying results could not be achieved.

A.4 Sequences return

An option for the GRU is to return its current state for each range step. Then, it is possible to concatenate these steps to create a vector, which may lead to an interesting prediction for the TL. It has been tested, and the conclusion was that not returning the sequences and keeping only the last state of the GRU unit is the best strategy for a reliable prediction.

B Prediction tool

The models obtained with the trainings on the presented datasets can be used to predict TLs, given wind profiles as input. The input format has to be specific in order to match the machine learning architectures. Currently, the wind profiles used must have the characteristics shown in Table 13.

Table 13: Model characteristics

type of model	height	number of height steps	range	space step	number of space steps
1000 km	120 km	1000	0-1000 km	100 km	11
4000 km	120 km	1000	0-4000 km	100 km	41

However, it should be easy to modify the input encoding process to adapt to new formats if necessary, for example, if the step is 50 km, as there is an interpolation to create the input.

C Error histograms

In this Section, we provide a thorough investigation of error distributions and error quantiles for each model and for varying ranges from the source and frequency bands.

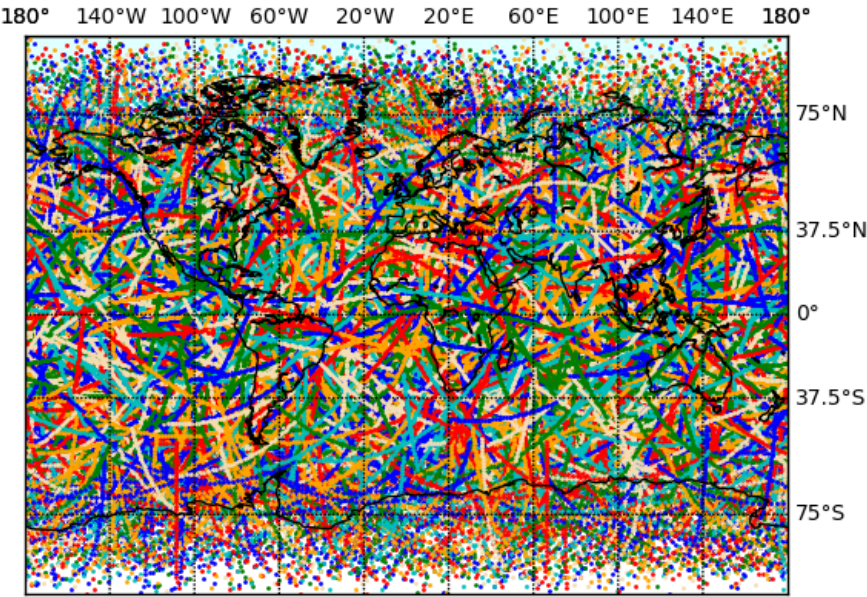


Figure 1: Distribution of slices used in the current study.

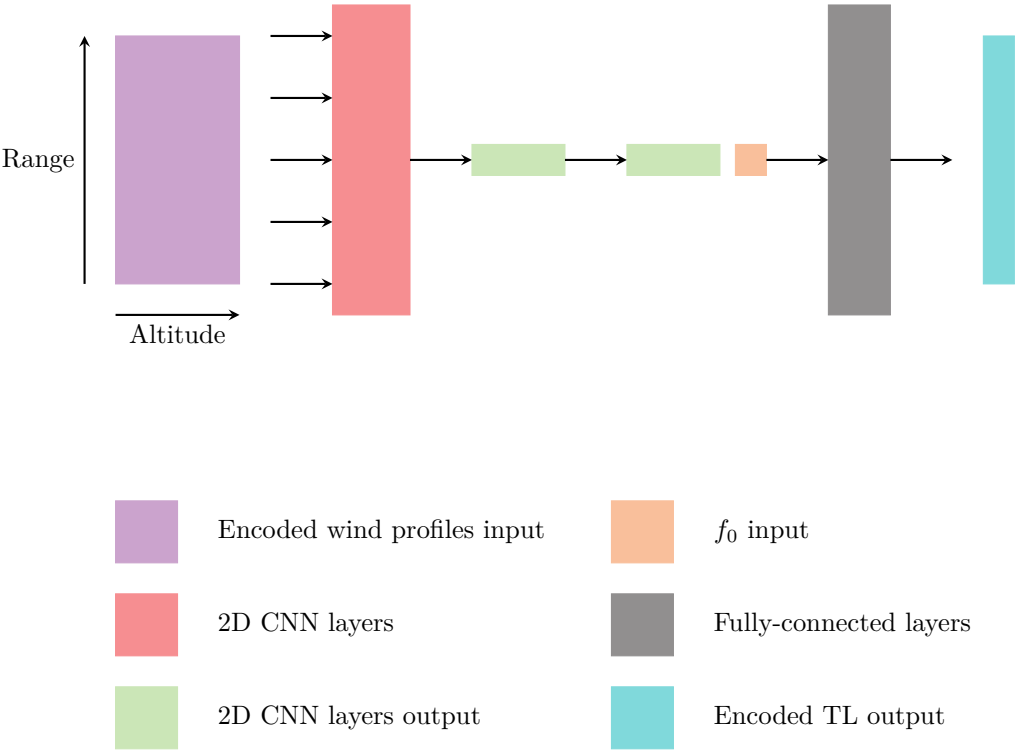


Figure 3: 2D CNN architecture

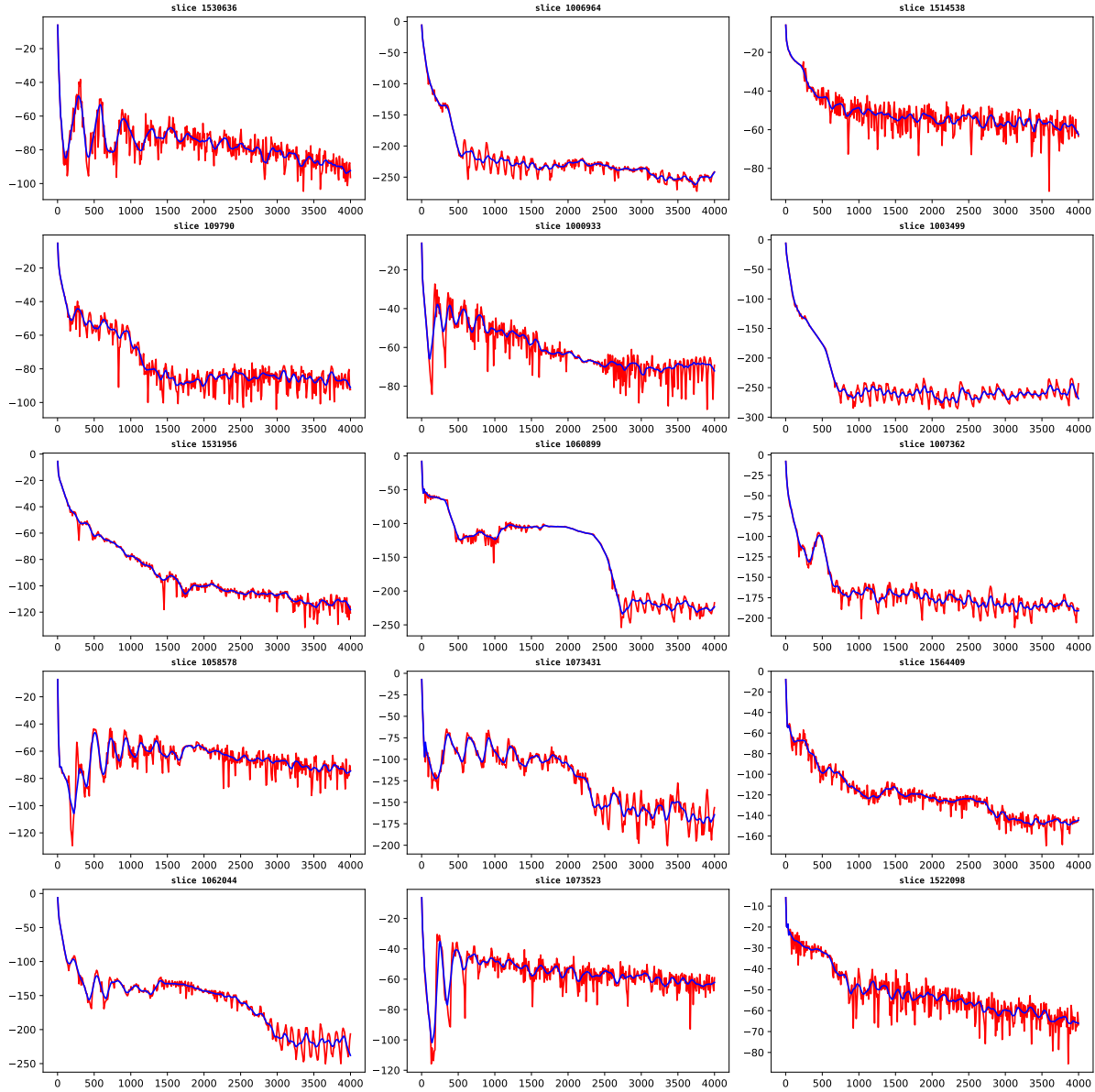


Figure 2: Original (red) and smoothed (blue) TL curves in db vs distances from the source in km for various atmospheric slices.

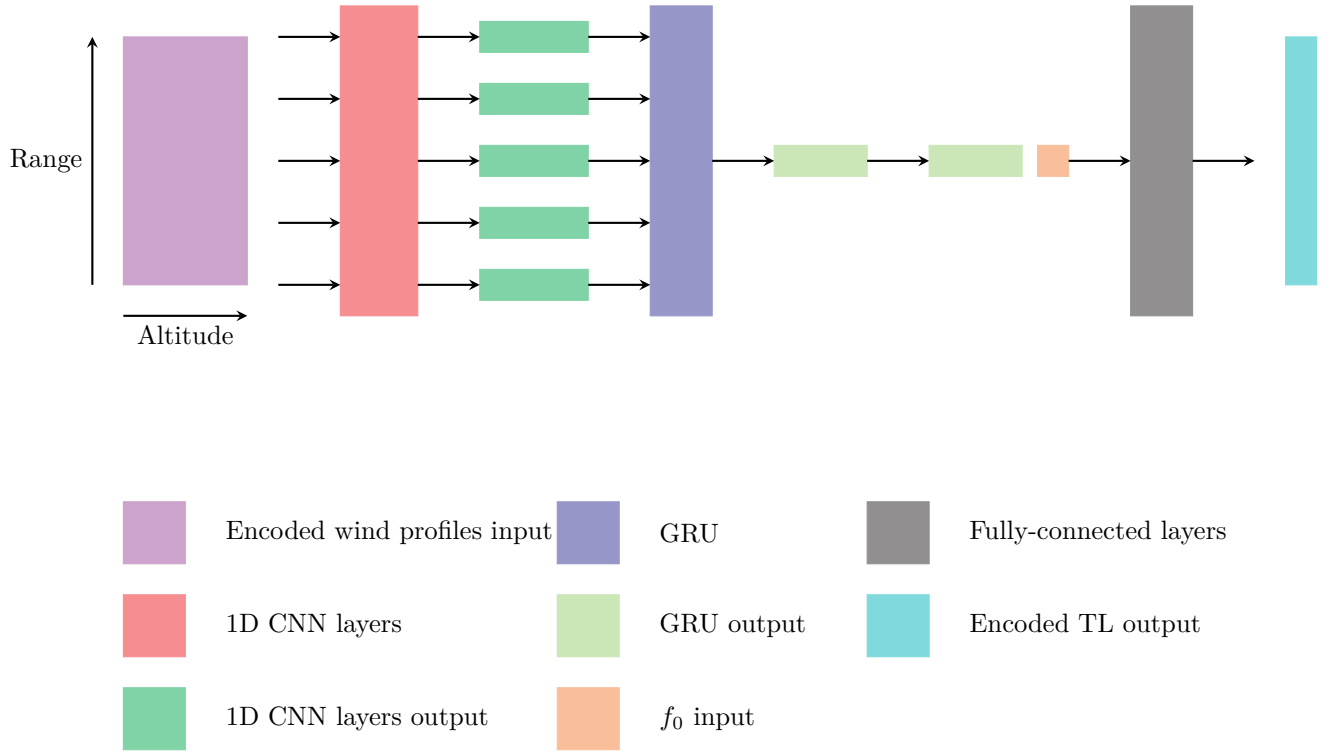


Figure 4: 1D CNN and GRU architecture

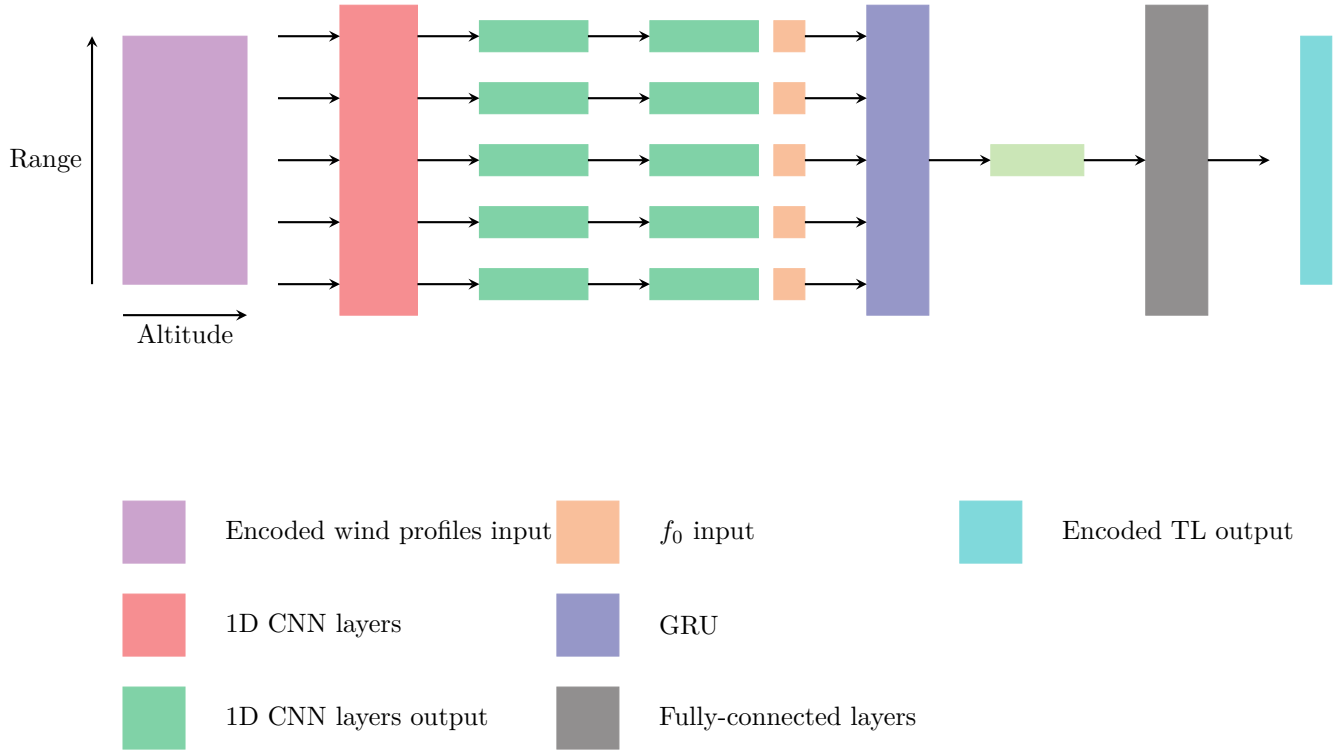


Figure 5: Final 1D CNN and GRU architecture

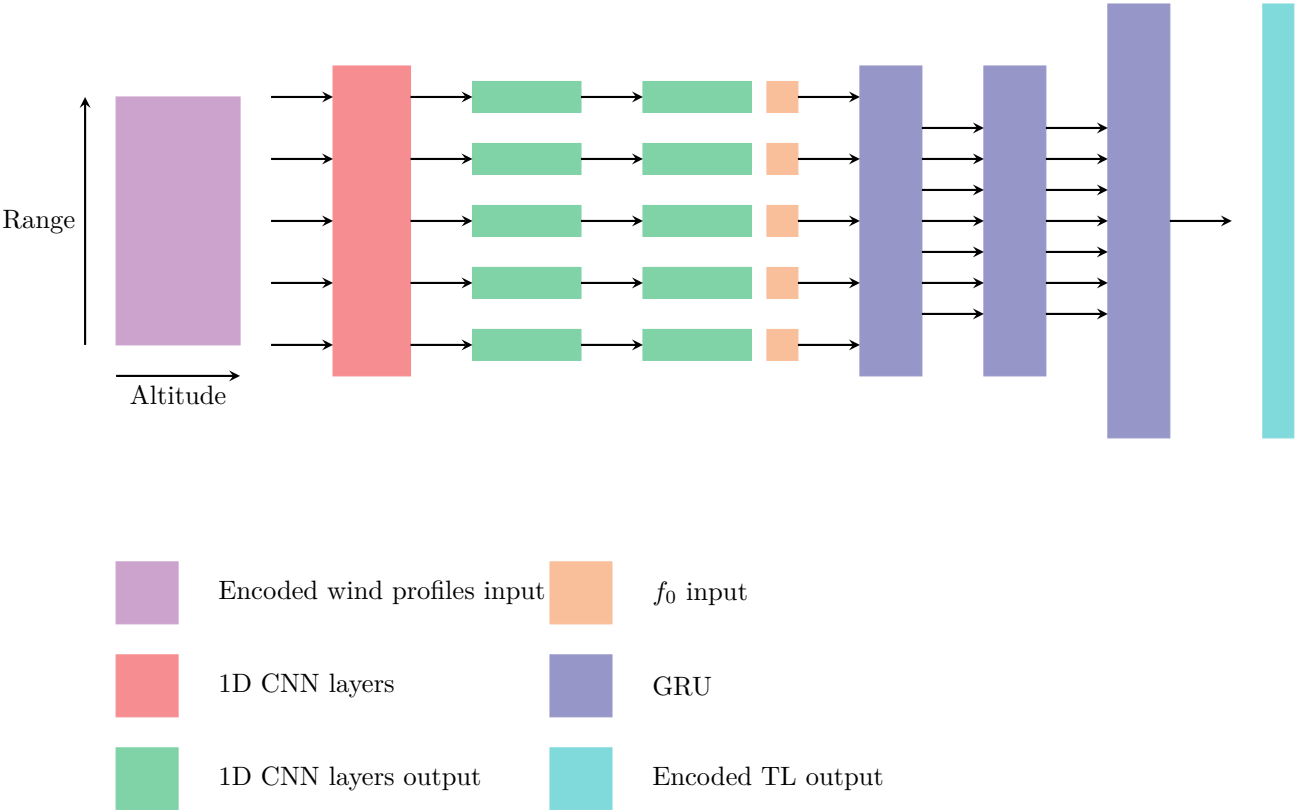


Figure 6: CNN and stacked GRUs architecture

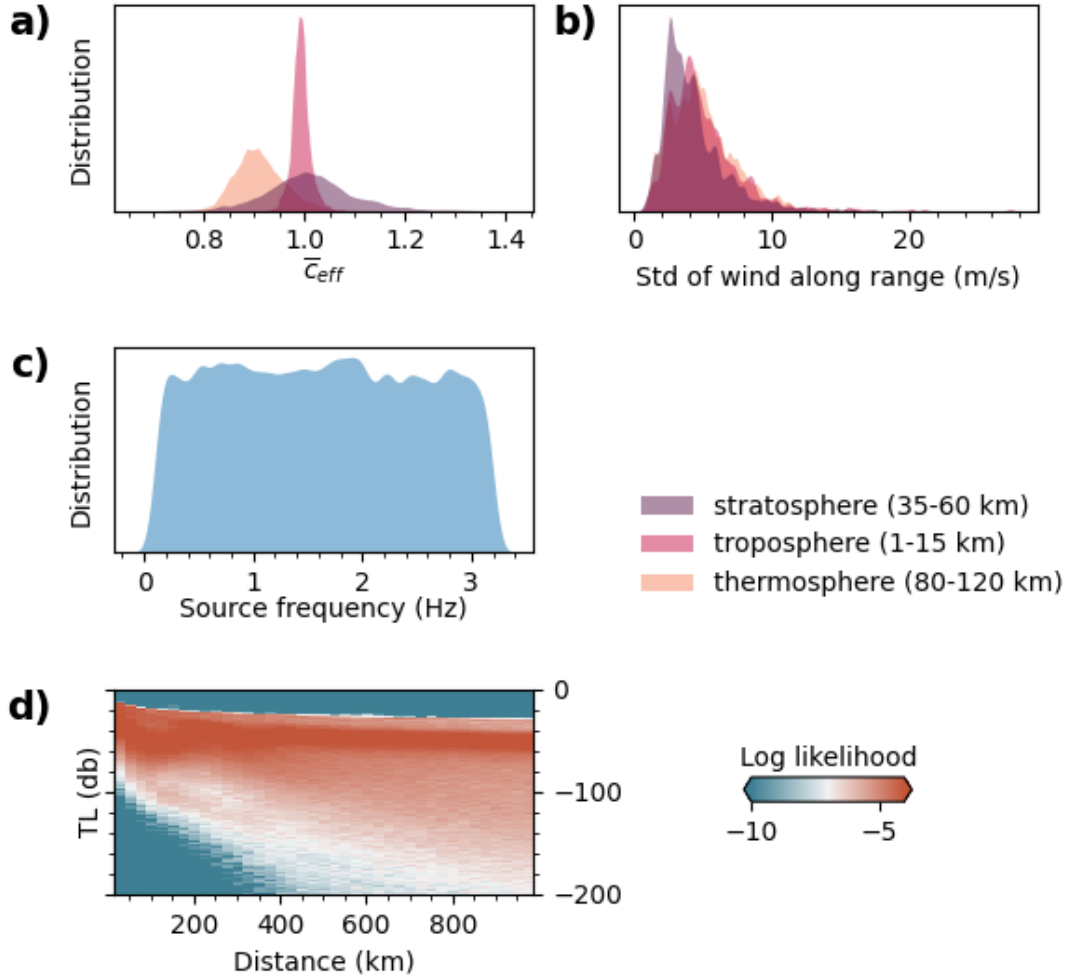


Figure 7: Parameter distribution in the 1000 km dataset. (a) Distribution of effective soundspeed ratio \bar{c}_{eff} between the ground and various atmospheric layers: troposphere (purple) between 1 and 15 km altitude, troposphere (purple) between 35 and 60 km altitude, and thermosphere (purple) between 80 and 120 km altitude. (b) Distribution of standard deviations of wind velocities along range for various atmospheric layers. (c) Distribution of input source frequencies used in PE simulations to build the entire TL dataset. (d) TL distribution represented as log likelihood (computed from Gaussian Kernel density estimates) vs distance determined from our entire TL dataset.

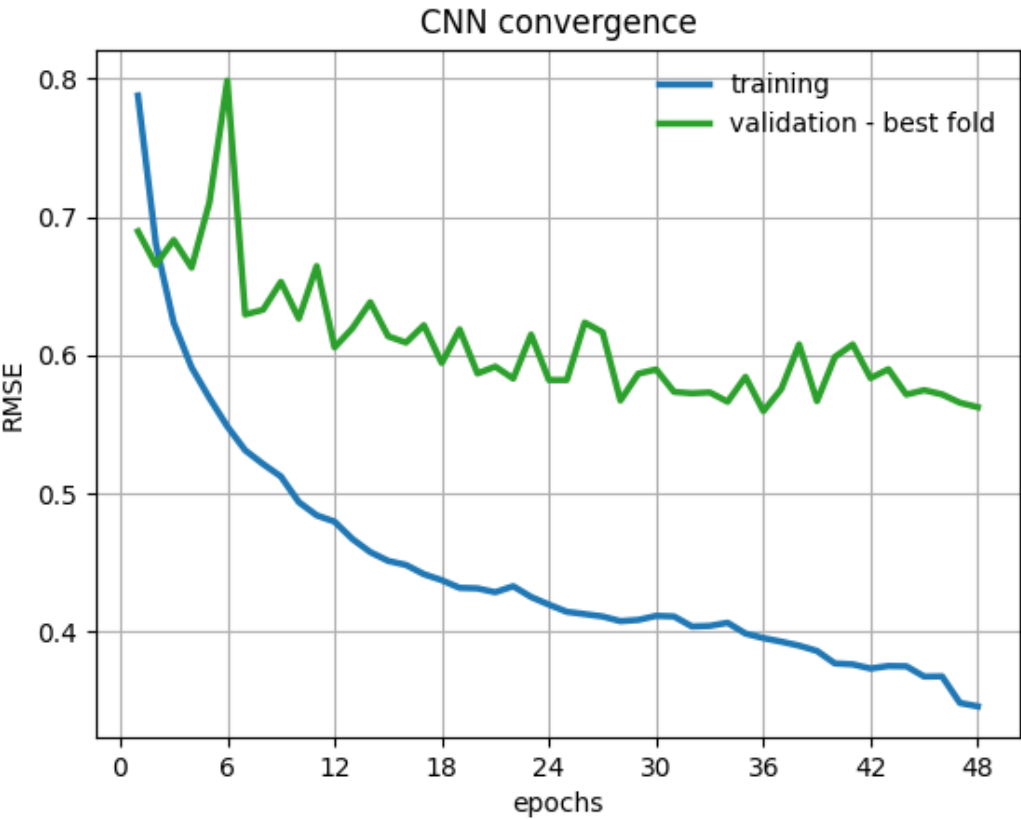


Figure 8: Convergence curves. Evolution of the RMSE with epochs of the 2D CNN model over the training (blue) and testing (green) datasets (1000 km dataset).

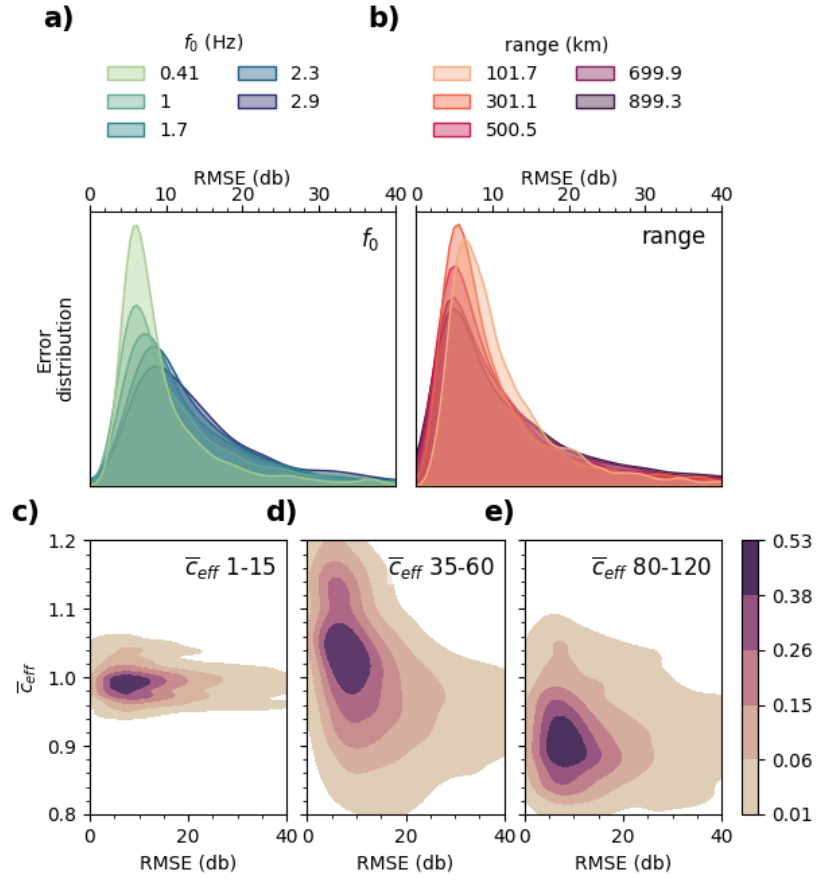
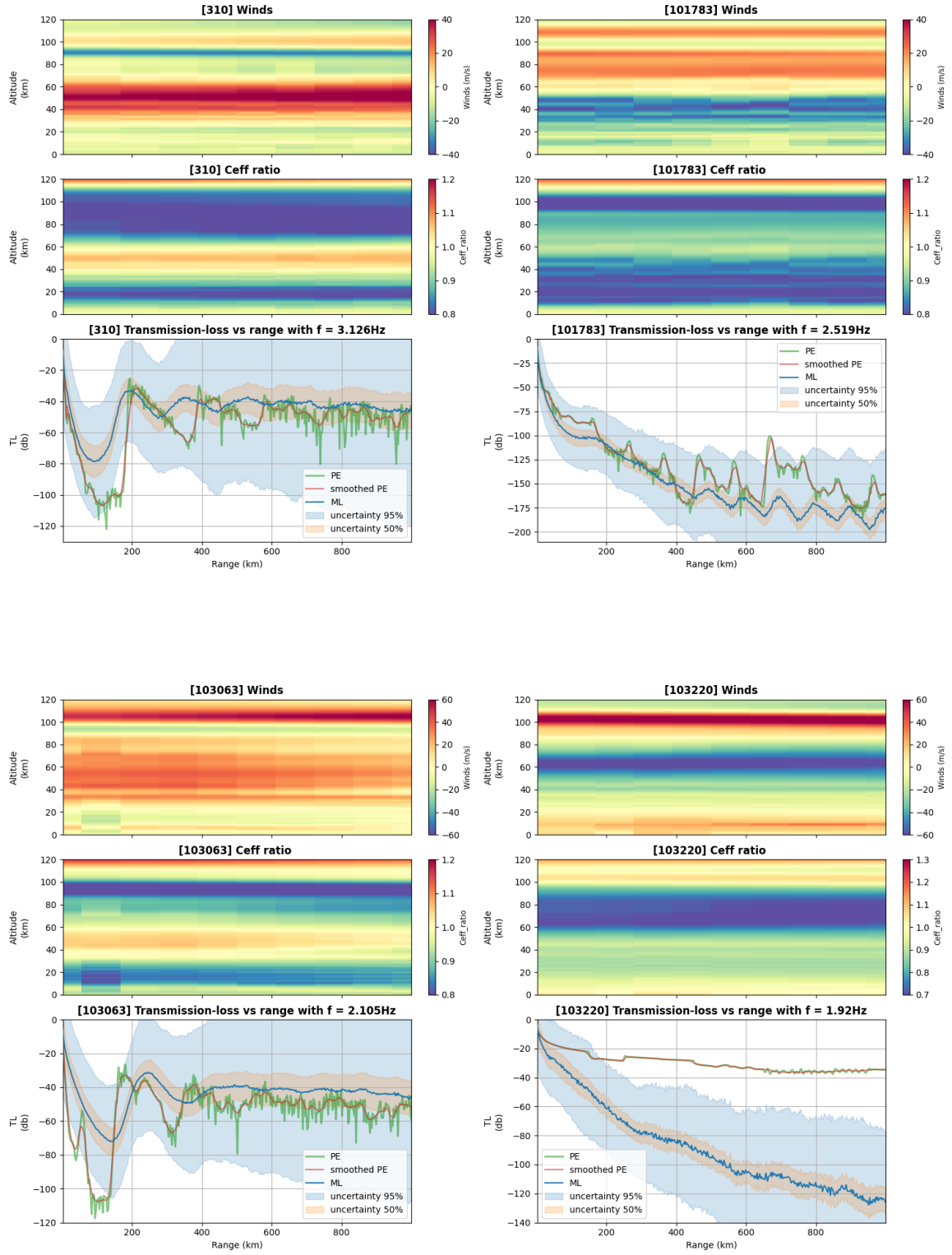


Figure 9: Error distributions for the 2D CNN model over the 1000 km dataset. (a) Distribution of RMSE over the testing dataset for various input frequencies. (b) Distribution of RMSE over the testing dataset for various ranges from the source. (c-e) Distribution of RMSE over the testing dataset for various values of effective soundspeed ratio \bar{c}_{eff} in (c) the troposphere, (d) the stratosphere, and (e) the thermosphere.



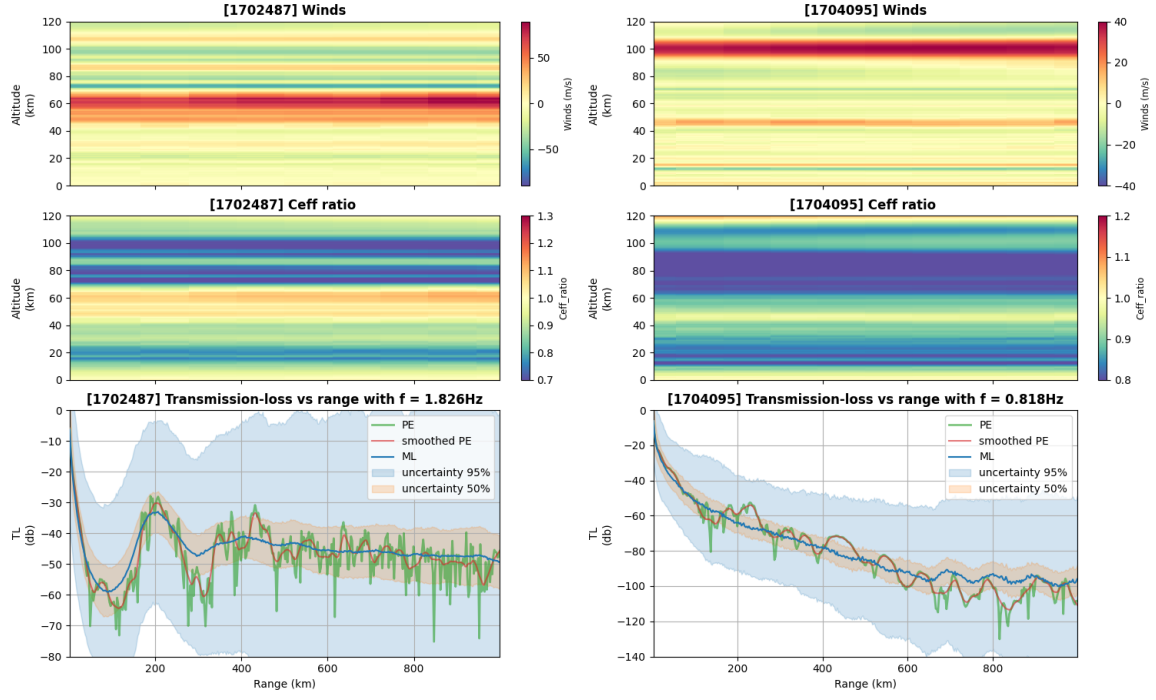


Figure 10: Comparisons for the 2D CNN model over the 1000 km dataset between the ML prediction (blue) plotted with its uncertainties (50% confidence and 95% confidence), the original PE results (green) and the smoothed PE results (red). The comparisons are plotted in TL (dB) versus range (km). The wind (m/s) and C_{eff} ratio profiles are shown above (in function of range and altitude in km) to better understand the atmosphere state and visualise the wave guides.

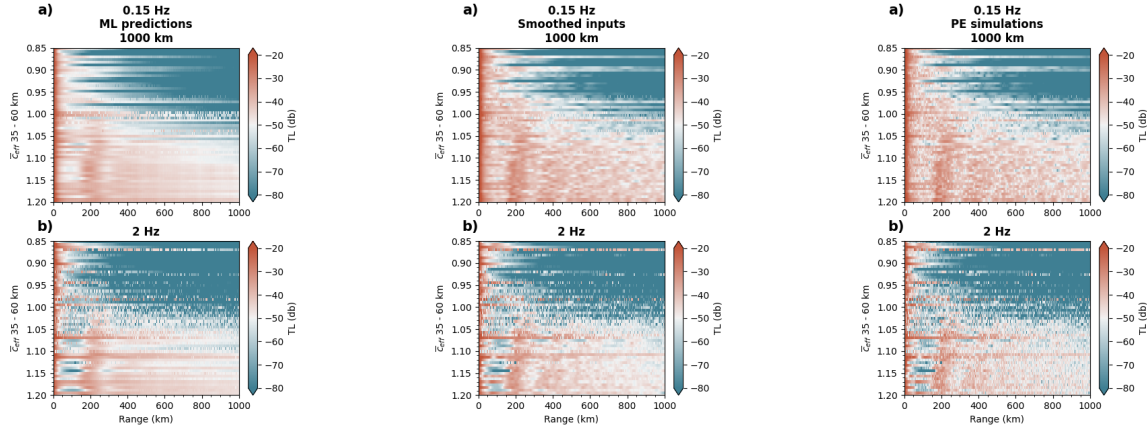


Figure 11: TL predictions (in db) for the 2D CNN model over the 1000 km dataset vs distance from the source (in km) and C_{eff} for the ML model (left), smoothed NCPA-simulated TL (middle), full NCPA-simulated TL (right) at 0.15 Hz (panels a and b) and 2 Hz (panels c and d).

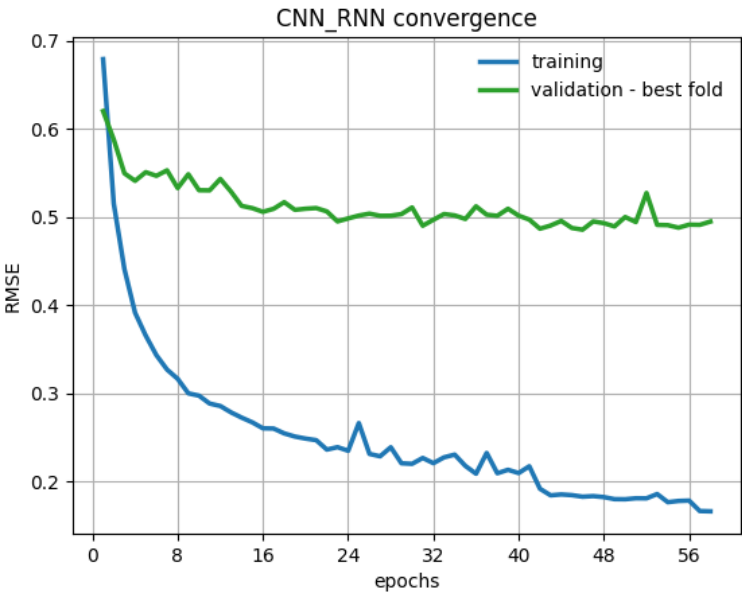


Figure 12: Convergence curves. Evolution of the RMSE with epochs of the 1D CNN-RNN model over the training (blue) and testing (green) datasets (1000 km dataset).

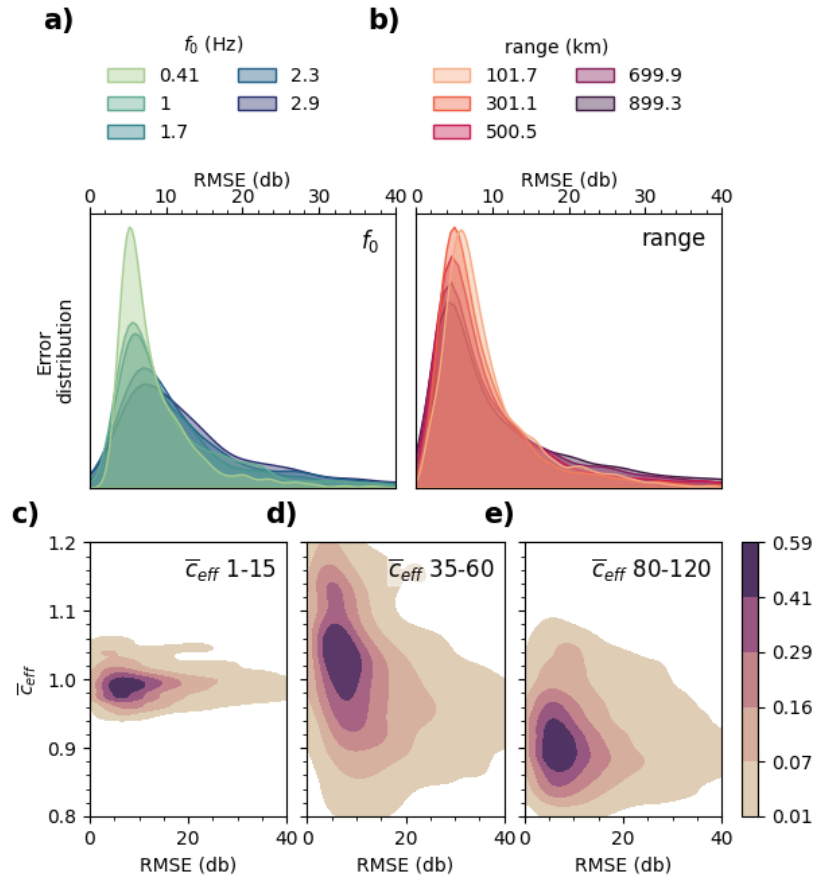
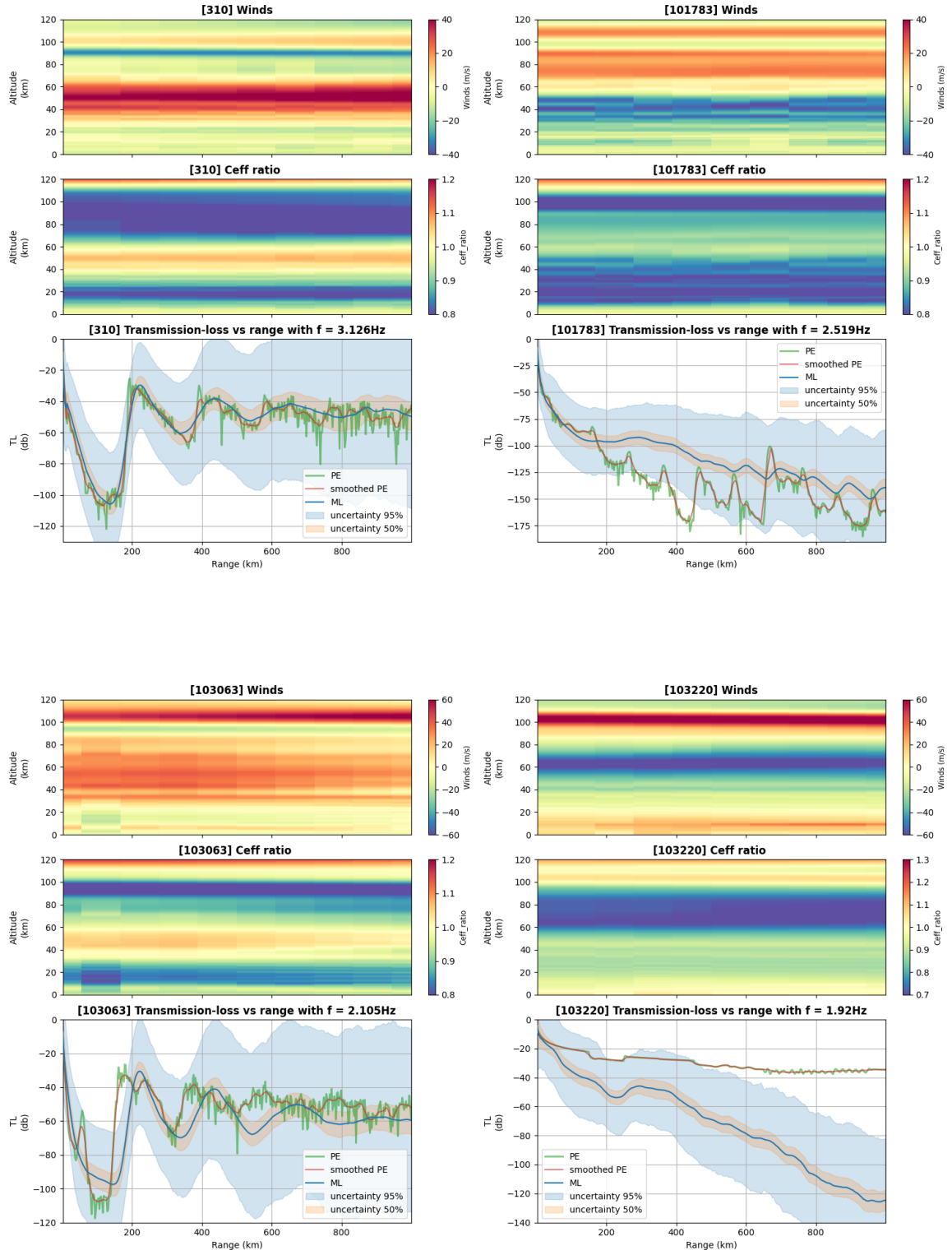


Figure 13: Error distributions for the 1D CNN-RNN model over the 1000 km dataset. (a) Distribution of RMSE over the testing dataset for various input frequencies. (b) Distribution of RMSE over the testing dataset for various ranges from the source. (c-e) Distribution of RMSE over the testing dataset for various values of effective sound speed ratio \bar{c}_{eff} in (c) the troposphere, (d) the stratosphere, and (e) the thermosphere.



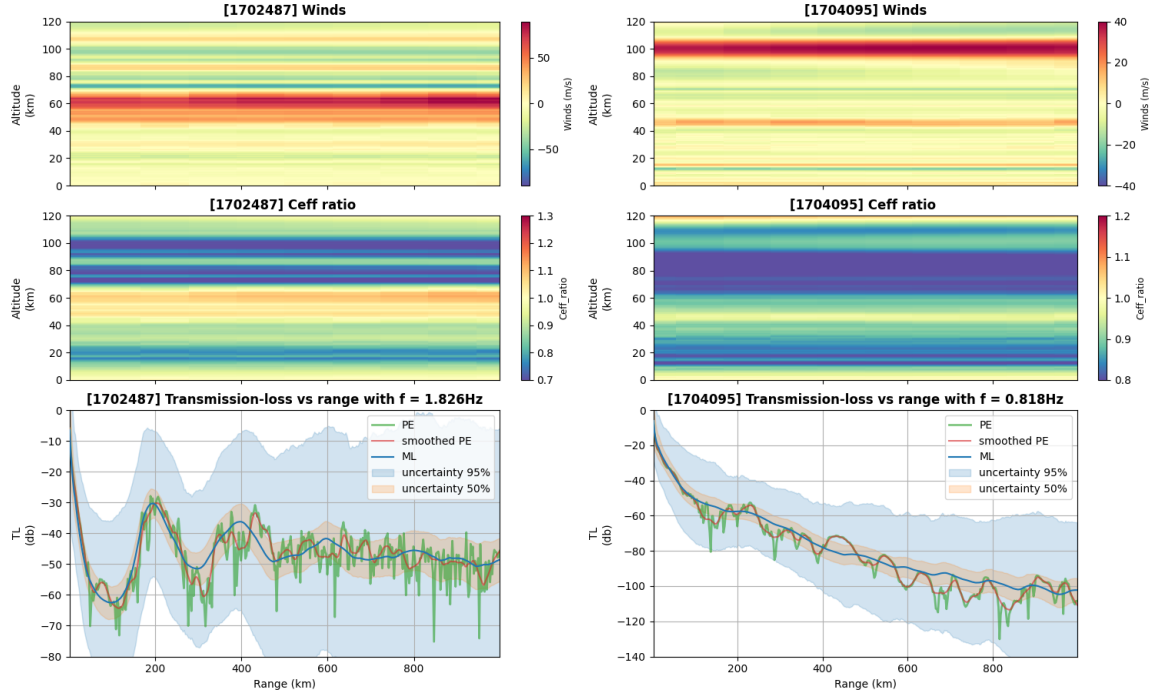


Figure 14: Comparisons for the 1D CNN-RNN model over the 1000 km dataset between the ML prediction (blue) plotted with its uncertainties (50% confidence and 95% confidence), the original PE results (green) and the smoothed PE results (red). The comparisons are plotted in TL (dB) versus range (km). The wind (m/s) and C_{eff} ratio profiles are shown above (in function of range and altitude in km) to better understand the atmosphere state and visualise the wave guides.

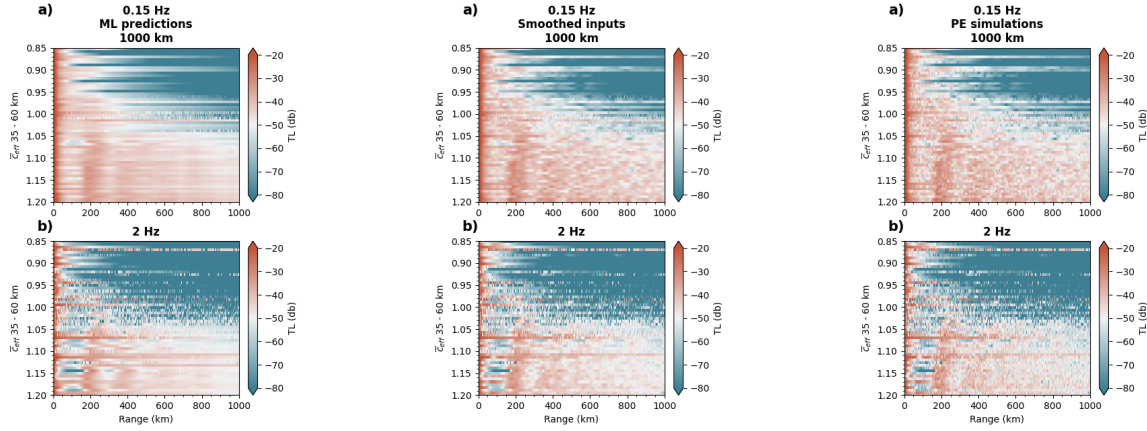


Figure 15: TL predictions (in db) for the 1D CNN-RNN model over the 1000 km dataset vs distance from the source (in km) and C_{eff} for the ML model (left), smoothed NCPA-simulated TL (middle), full NCPA-simulated TL (right) at 0.15 Hz (panels a and b) and 2 Hz (panels c and d).

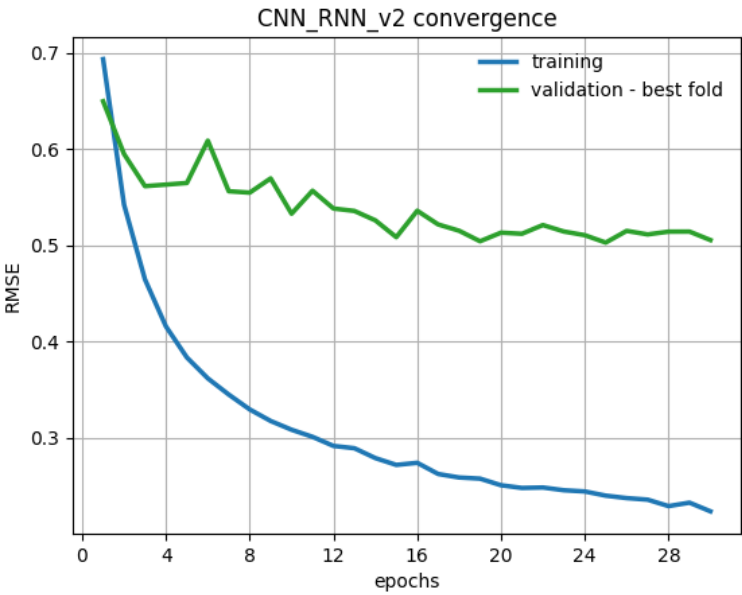


Figure 16: Convergence curves. Evolution of the RMSE with epochs of the 1D CNN-RNN with stacked GRUs model (GRU of size 110) over the training (blue) and testing (green) datasets (1000 km dataset).

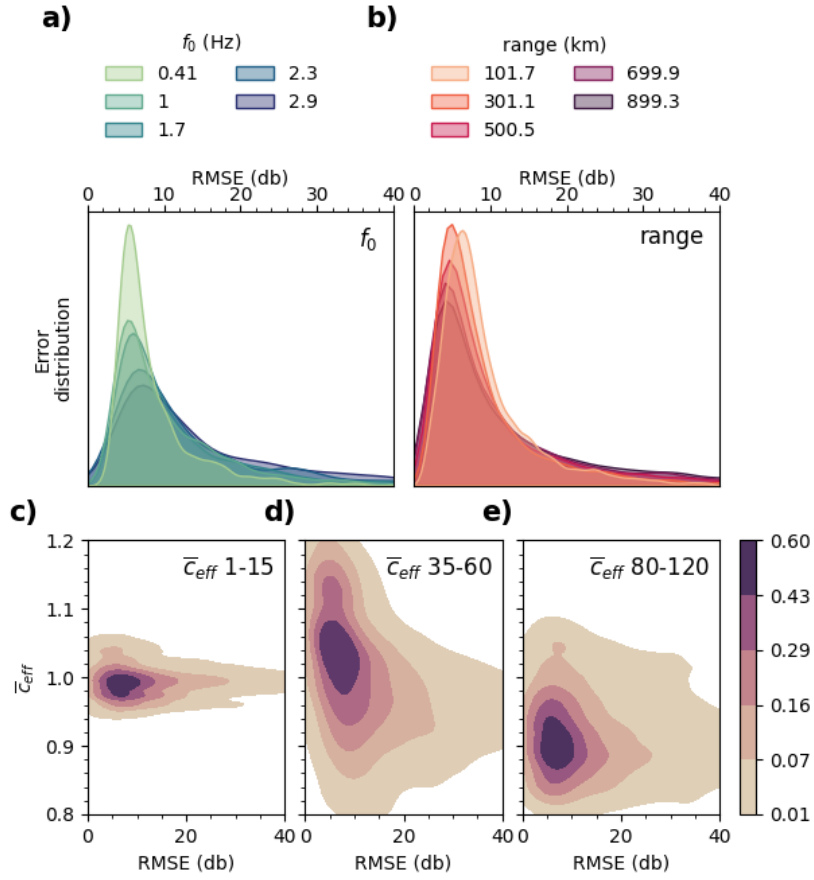
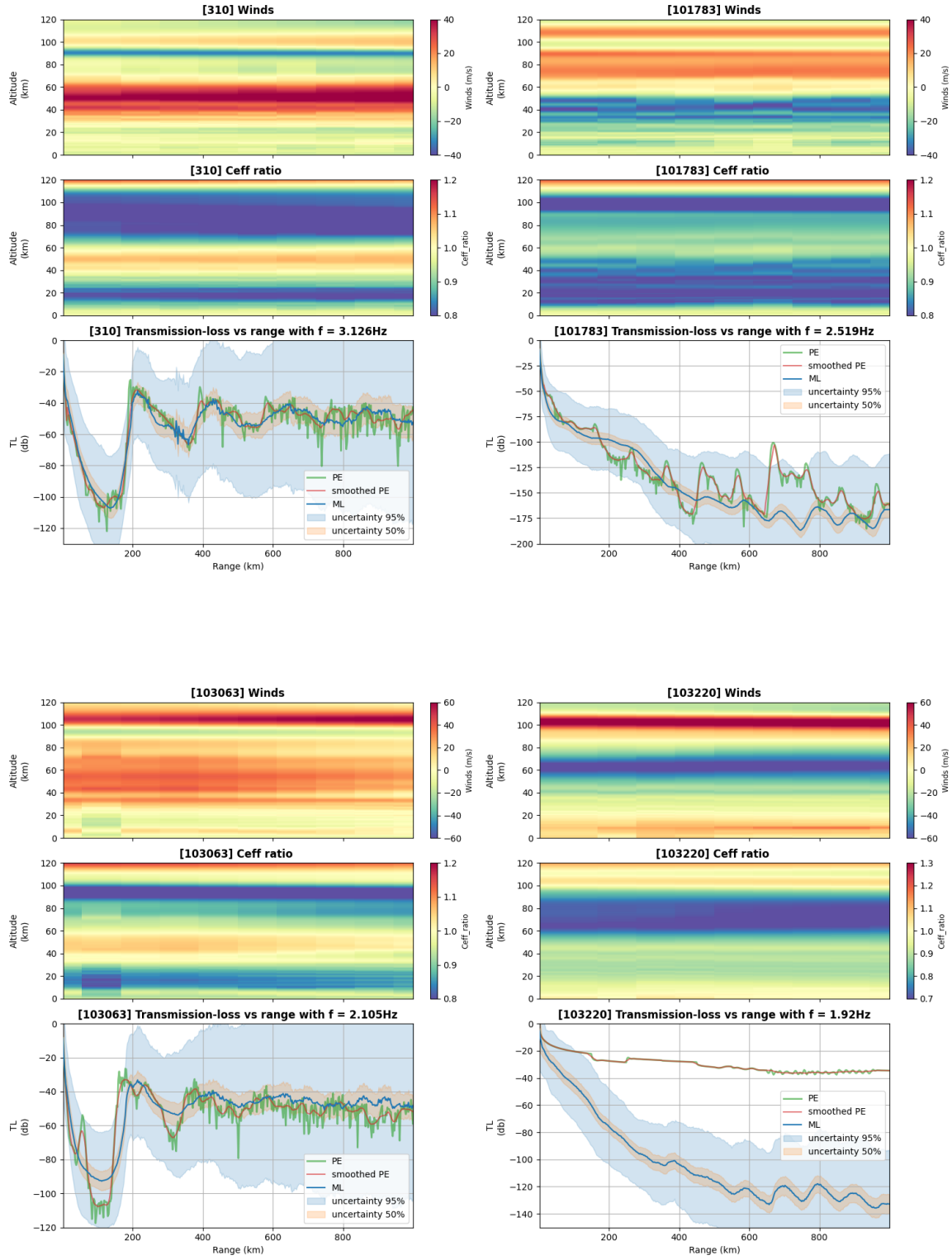


Figure 17: Error distributions for the 1D CNN-RNN with stacked GRUs model (GRU of size 110) over the 1000 km dataset. (a) Distribution of RMSE over the testing dataset for various input frequencies. (b) Distribution of RMSE over the testing dataset for various ranges from the source. (c-e) Distribution of RMSE over the testing dataset for various values of effective soundspeed ratio \bar{c}_{eff} in (c) the troposphere, (d) the stratosphere, and (e) the thermosphere.



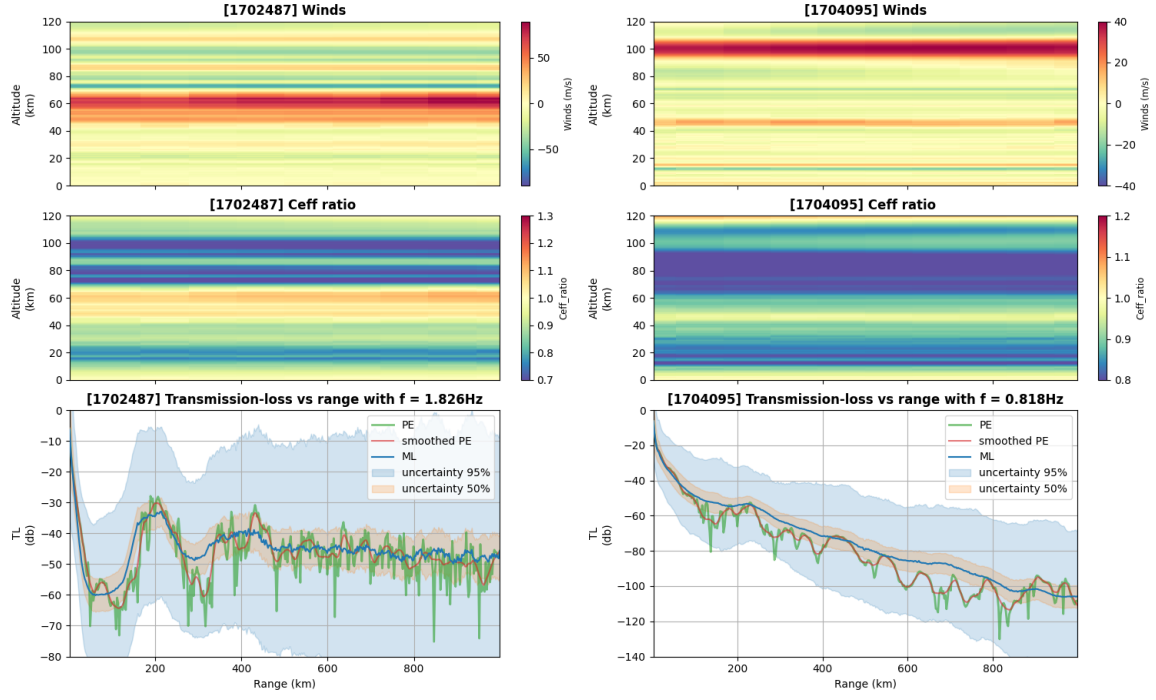


Figure 18: Comparisons for the 1D CNN-RNN with stacked GRUs model (GRU of size 110) over the 1000 km dataset between the ML prediction (blue) plotted with its uncertainties (50% confidence and 95% confidence), the original PE results (green) and the smoothed PE results (red). The comparisons are plotted in TL (dB) versus range (km). The wind (m/s) and C_{eff} ratio profiles are shown above (in function of range and altitude in km) to better understand the atmosphere state and visualise the wave guides.

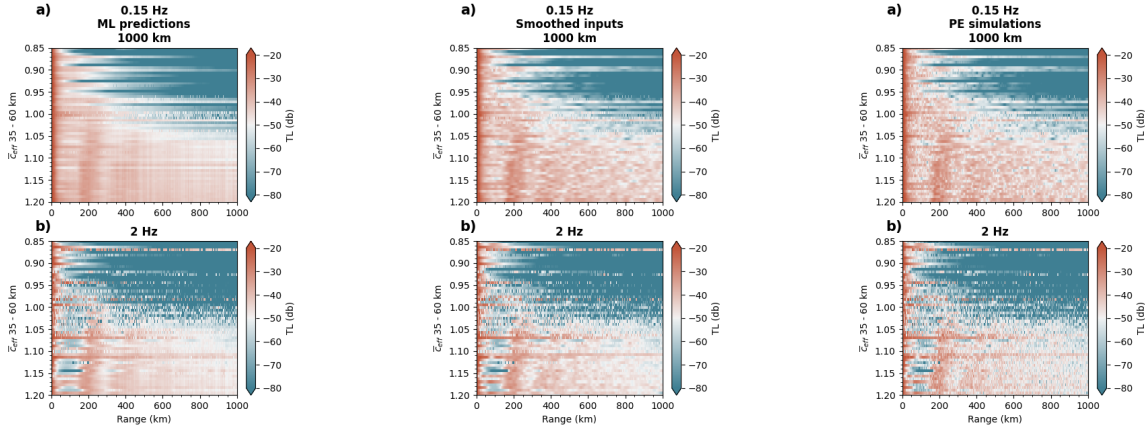


Figure 19: TL predictions (in db) for the 1D CNN-RNN with stacked GRUs model (GRU of size 110) over the 1000 km dataset vs distance from the source (in km) and C_{eff} for the ML model (left), smoothed NCPA-simulated TL (middle), full NCPA-simulated TL (right) at 0.15 Hz (panels a and b) and 2 Hz (panels c and d).

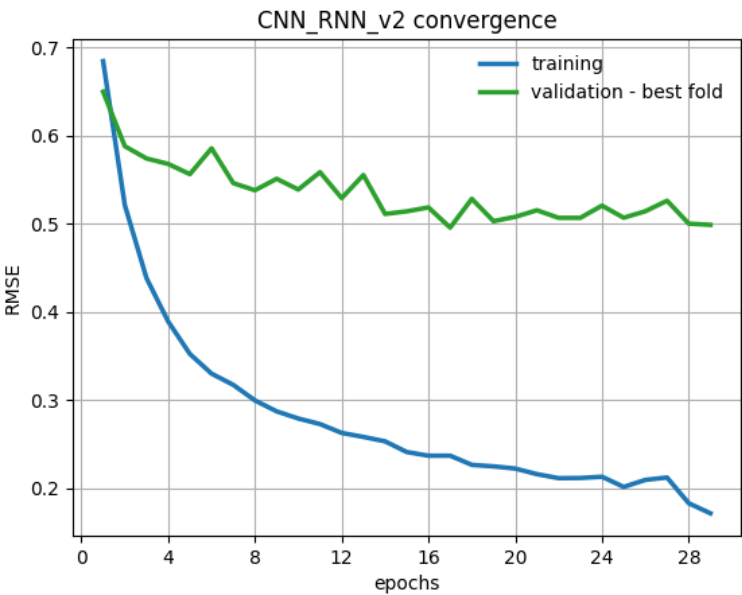


Figure 20: Convergence curves. Evolution of the RMSE with epochs of the 1D CNN-RNN with stacked GRUs model (GRU of size 400) over the training (blue) and testing (green) datasets (1000 km dataset).

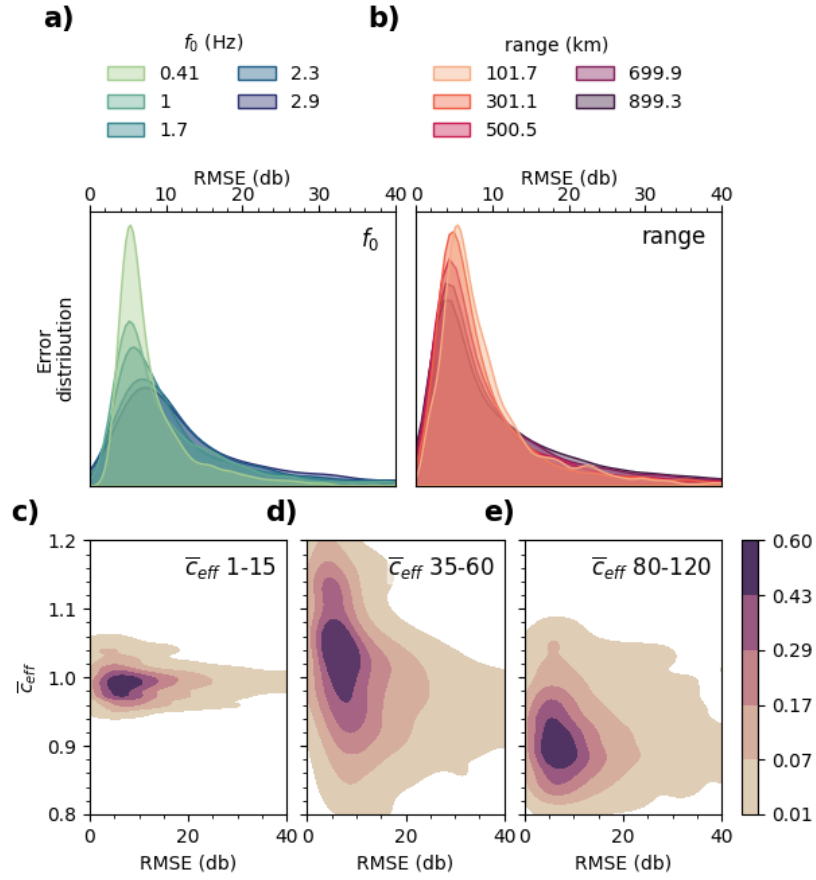
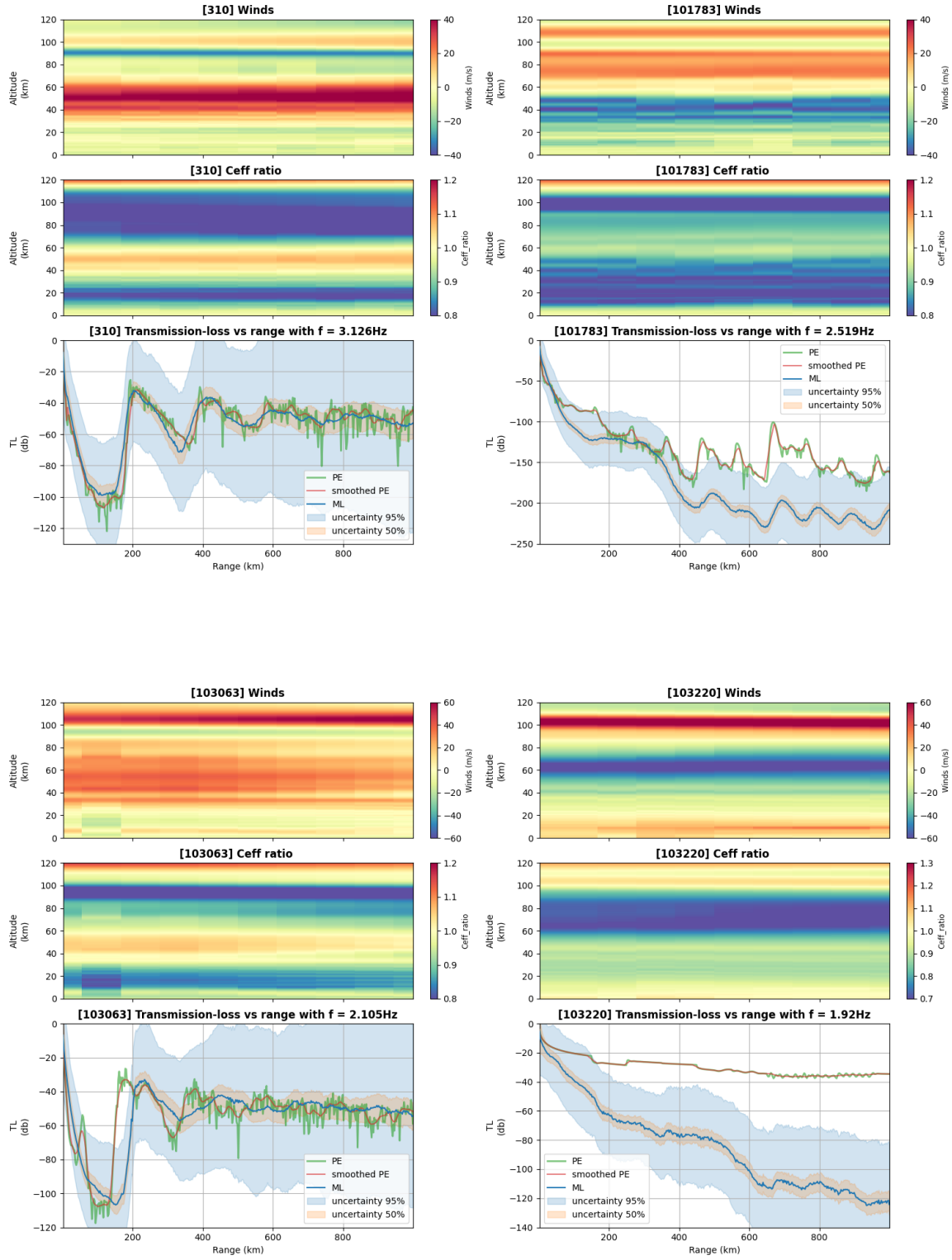


Figure 21: Error distributions for the 1D CNN-RNN with stacked GRUs model (GRU of size 400) over the 1000 km dataset. (a) Distribution of RMSE over the testing dataset for various input frequencies. (b) Distribution of RMSE over the testing dataset for various ranges from the source. (c-e) Distribution of RMSE over the testing dataset for various values of effective soundspeed ratio \bar{c}_{eff} in (c) the troposphere, (d) the stratosphere, and (e) the thermosphere.



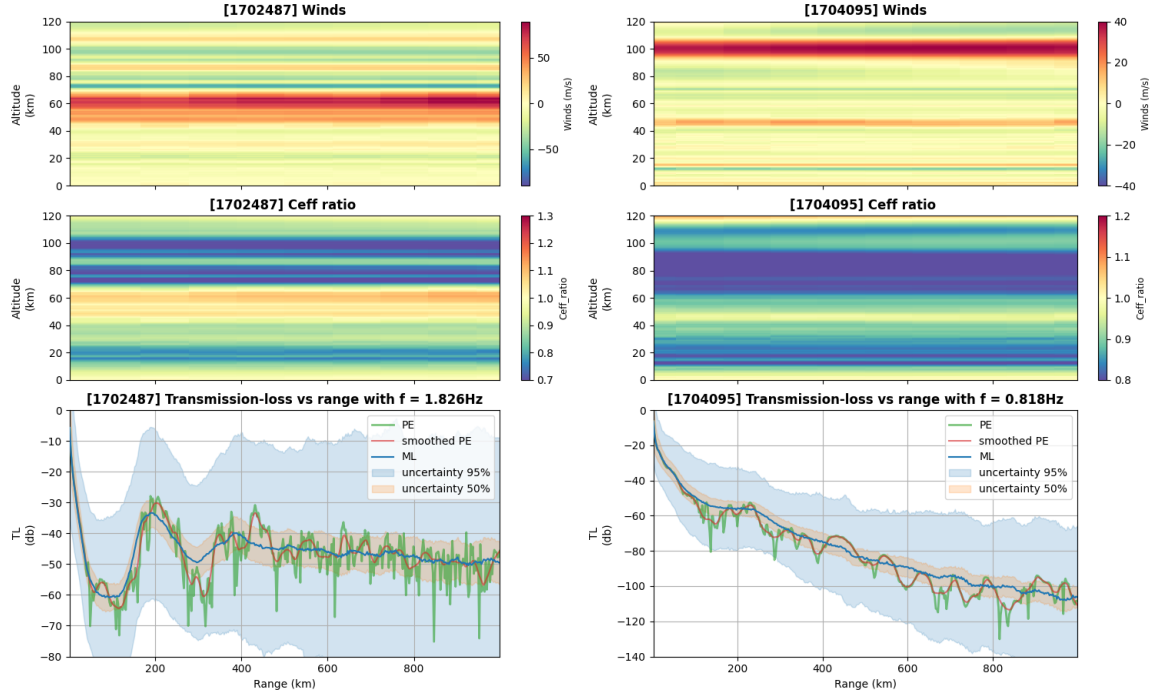


Figure 22: Comparisons for the 1D CNN-RNN with stacked GRUs model (GRU of size 400) over the 1000 km dataset between the ML prediction (blue) plotted with its uncertainties (50% confidence and 95% confidence), the original PE results (green) and the smoothed PE results (red). The comparisons are plotted in TL (dB) versus range (km). The wind (m/s) and C_{eff} ratio profiles are shown above (in function of range and altitude in km) to better understand the atmosphere state and visualise the wave guides.

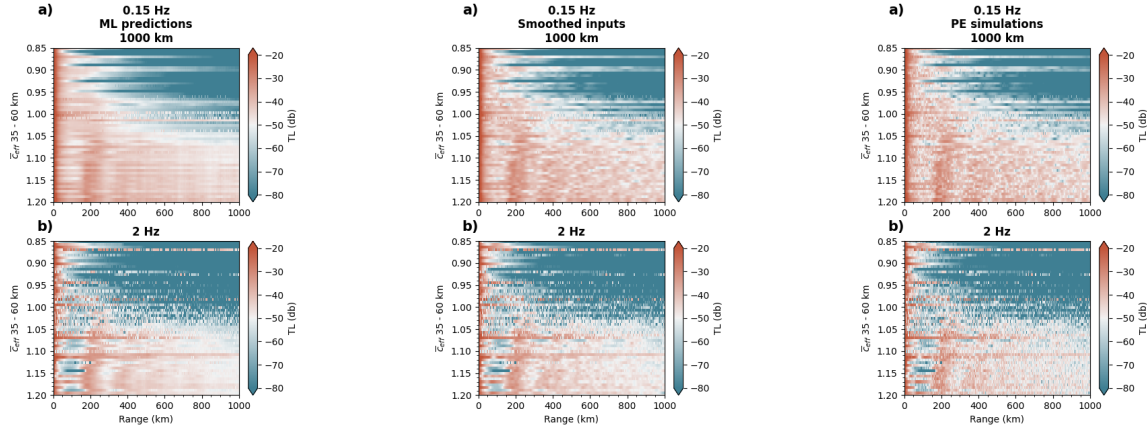


Figure 23: TL predictions (in db) for the 1D CNN-RNN with stacked GRUs model (GRU of size 400) over the 1000 km dataset vs distance from the source (in km) and C_{eff} for the ML model (left), smoothed NCPA-simulated TL (middle), full NCPA-simulated TL (right) at 0.15 Hz (panels a and b) and 2 Hz (panels c and d).

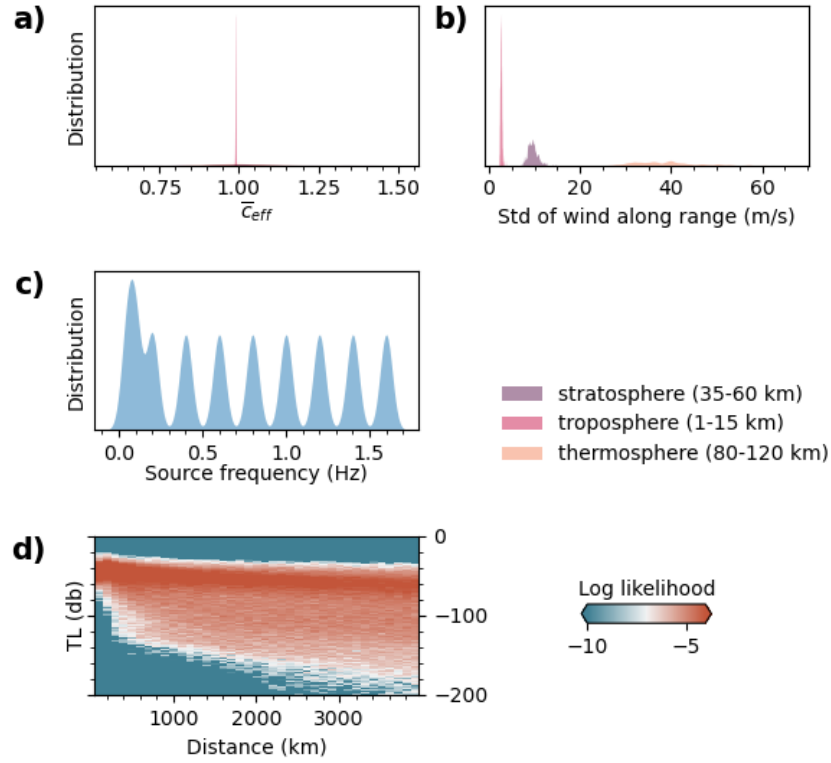


Figure 24: Parameter distribution in the 4000 km dataset from Alexis Le Pichon. (a) Distribution of effective soundspeed ratio \bar{c}_{eff} between the ground and various atmospheric layers: troposphere (purple) between 1 and 15 km altitude, troposphere (purple) between 35 and 60 km altitude, and thermosphere (purple) between 80 and 120 km altitude. (b) Distribution of standard deviations of wind velocities along range for various atmospheric layers. (c) Distribution of input source frequencies used in PE simulations to build the entire TL dataset. (d) TL distribution represented as log likelihood (computed from Gaussian Kernel density estimates) vs distance determined from our entire TL dataset.

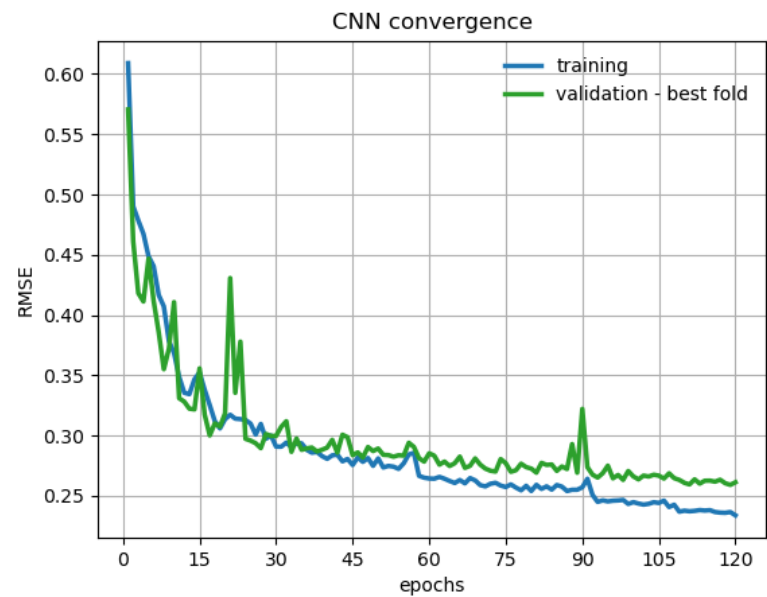


Figure 25: Convergence curves. Evolution of the RMSE with training epochs of the 2D CNN model over the training (blue) and testing (green) datasets (4000 km dataset from Alexis Le Pichon).

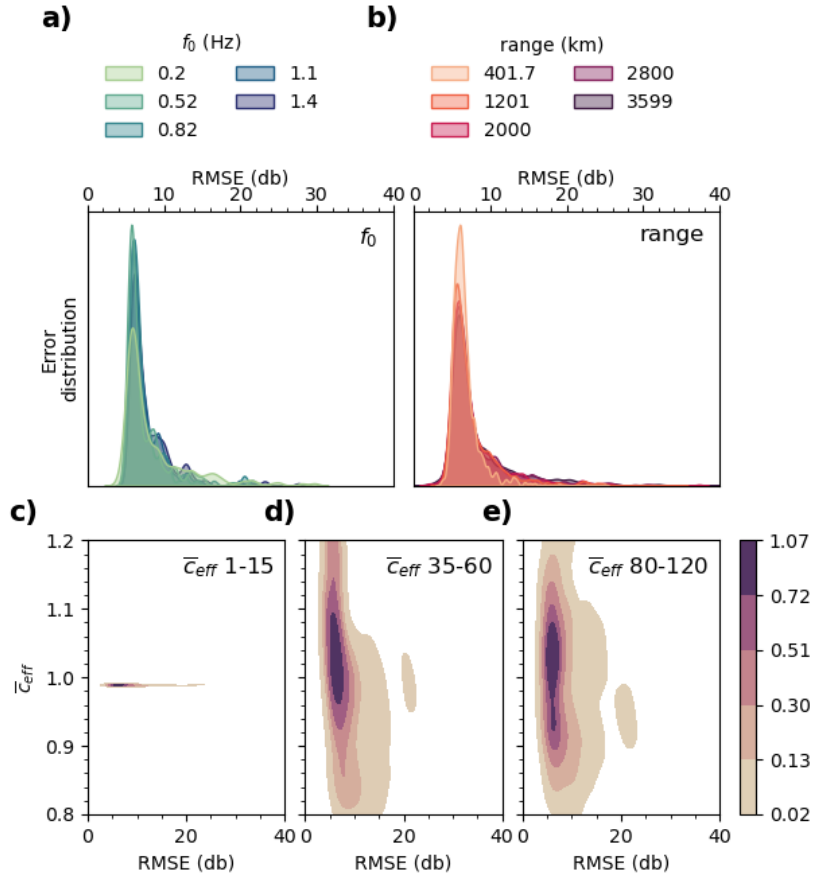
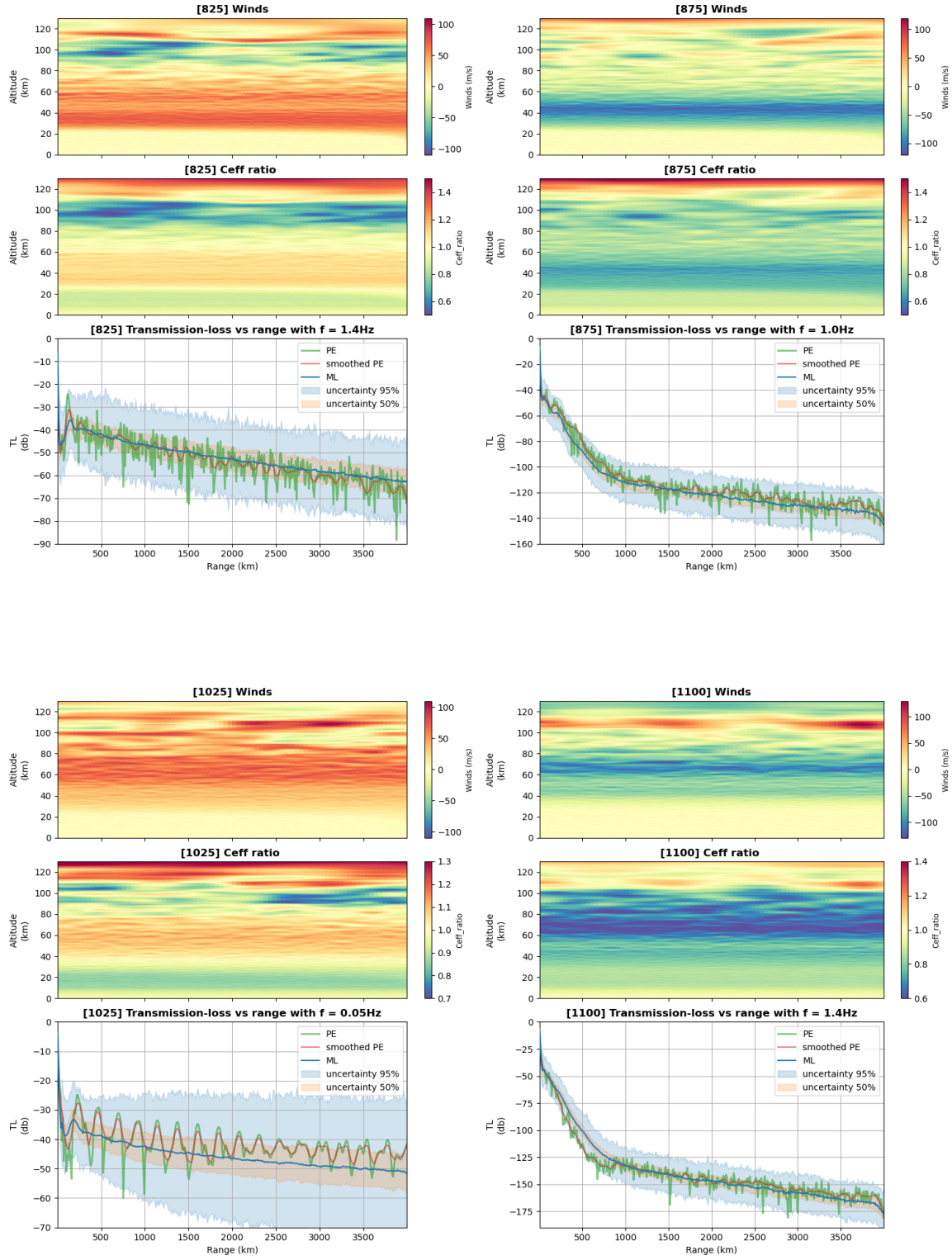


Figure 26: Error distributions for the 2D CNN model over the 4000 km dataset from Alexis Le Pichon. (a) Distribution of RMSE over the testing dataset for various input frequencies. (b) Distribution of RMSE over the testing dataset for various ranges from the source. (c-e) Distribution of RMSE over the testing dataset for various values of effective soundspeed ratio \bar{c}_{eff} in (c) the troposphere, (d) the stratosphere, and (e) the thermosphere.



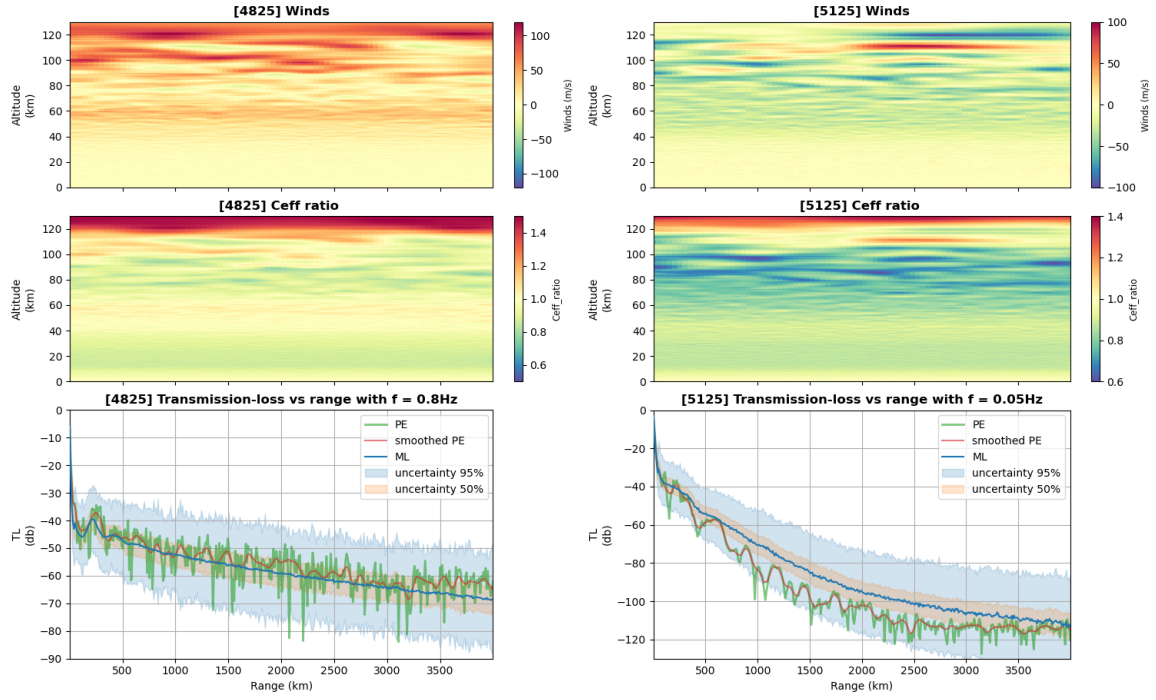


Figure 27: Comparisons for the 2D CNN model over the 4000 km dataset from Alexis Le Pichon between the ML prediction (blue) plotted with its uncertainties (50% confidence and 95% confidence), the original PE results (green) and the smoothed PE results (red). The comparisons are plotted in TL (dB) versus range (km). The wind (m/s) and C_{eff} ratio profiles are shown above (in function of range and altitude in km) to better understand the atmosphere state and visualise the wave guides.

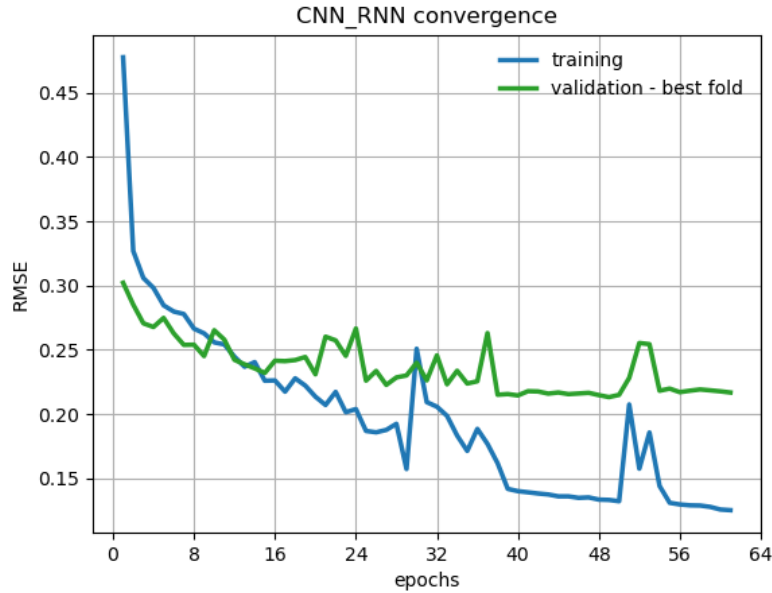


Figure 28: Convergence curves. Evolution of the RMSE with epochs of the 1D CNN-RNN model over the training (blue) and testing (green) datasets (4000 km dataset from Alexis Le Pichon).

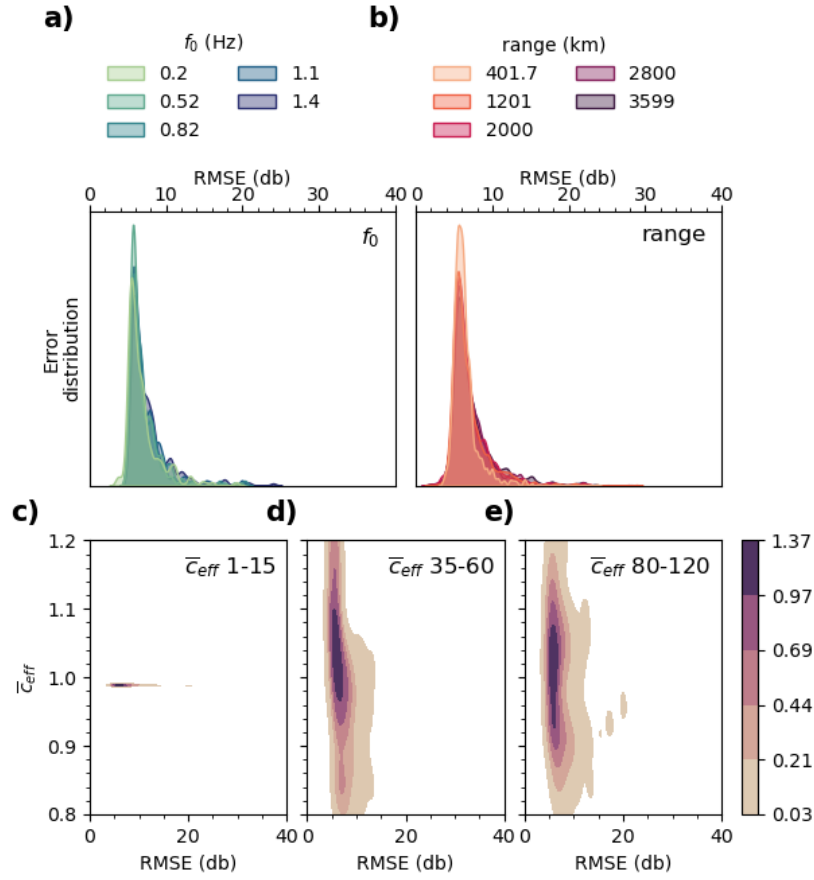
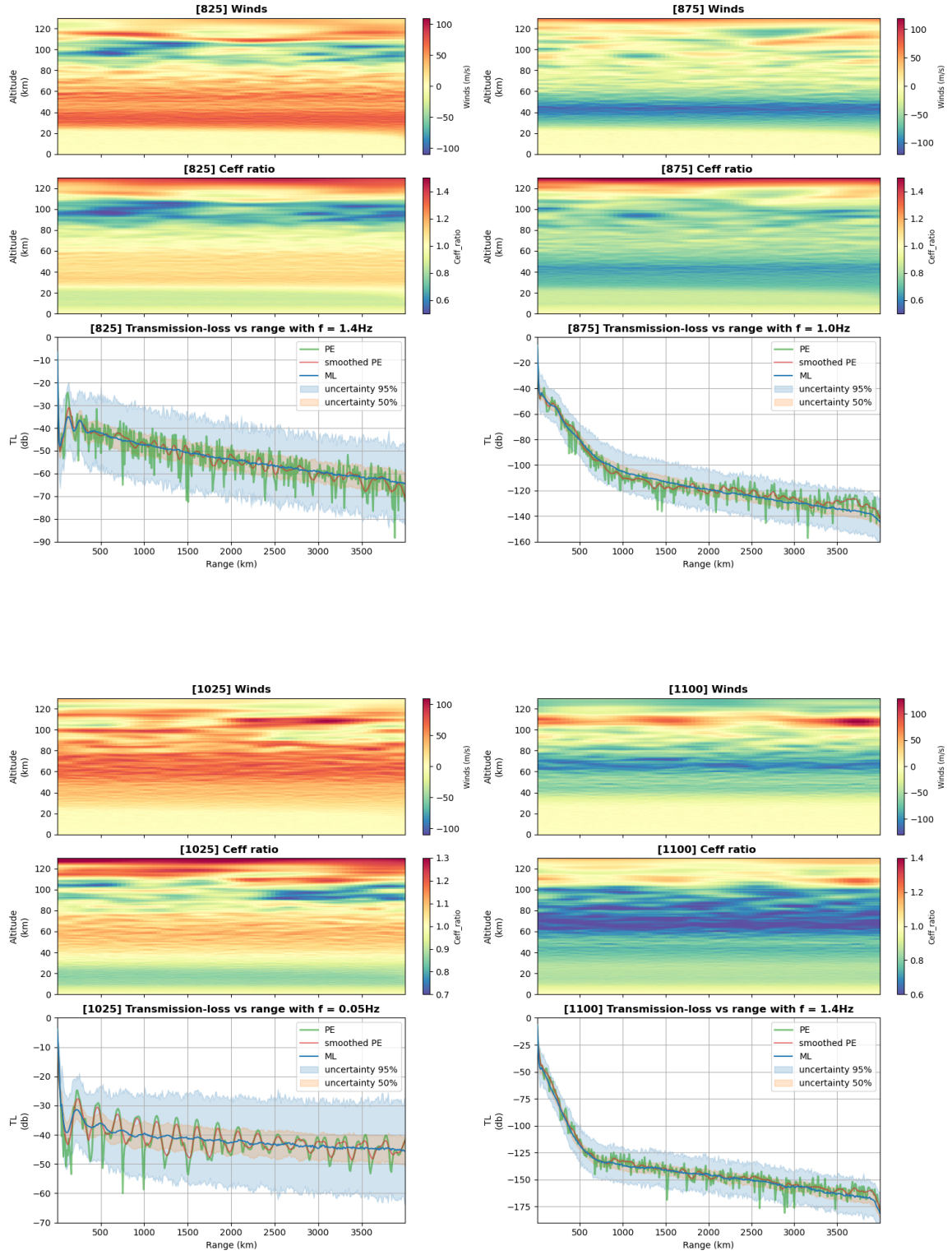


Figure 29: Error distributions for the 1D CNN-RNN model over the 4000 km dataset from Alexis Le Pichon. (a) Distribution of RMSE over the testing dataset for various input frequencies. (b) Distribution of RMSE over the testing dataset for various ranges from the source. (c-e) Distribution of RMSE over the testing dataset for various values of effective soundspeed ratio \bar{c}_{eff} in (c) the troposphere, (d) the stratosphere, and (e) the thermosphere.



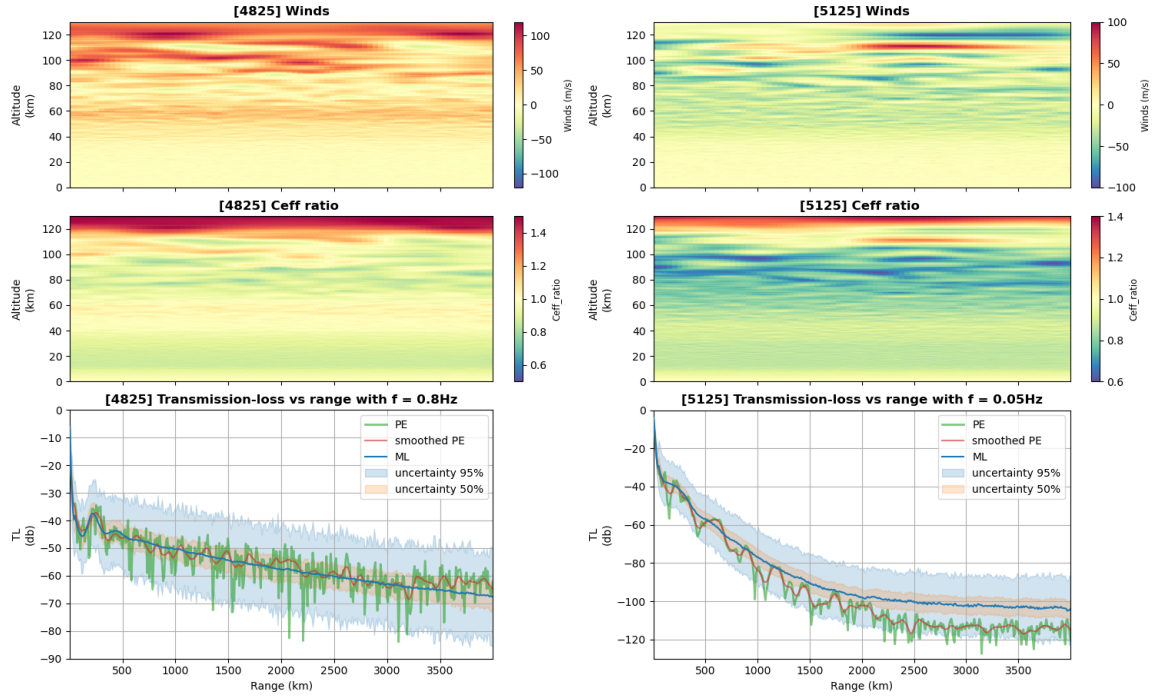


Figure 30: Comparisons for the 1D CNN-RNN model over the 4000 km dataset from Alexis Le Pichon between the ML prediction (blue) plotted with its uncertainties (50% confidence and 95% confidence), the original PE results (green) and the smoothed PE results (red). The comparisons are plotted in TL (dB) versus range (km). The wind (m/s) and C_{eff} ratio profiles are shown above (in function of range and altitude in km) to better understand the atmosphere state and visualise the wave guides.

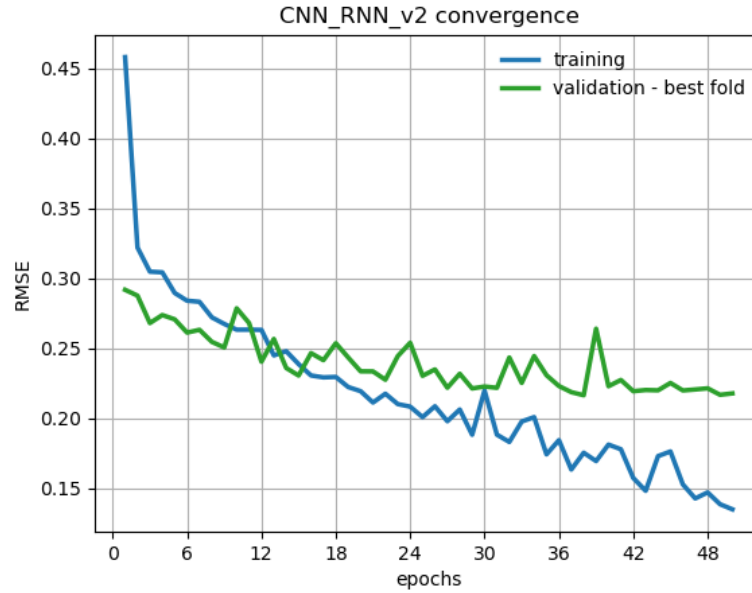


Figure 31: Convergence curves. Evolution of the RMSE with epochs of the 1D CNN-RNN with stacked GRUs model (GRU of size 110) over the training (blue) and testing (green) datasets (4000 km dataset from Alexis Le Pichon).

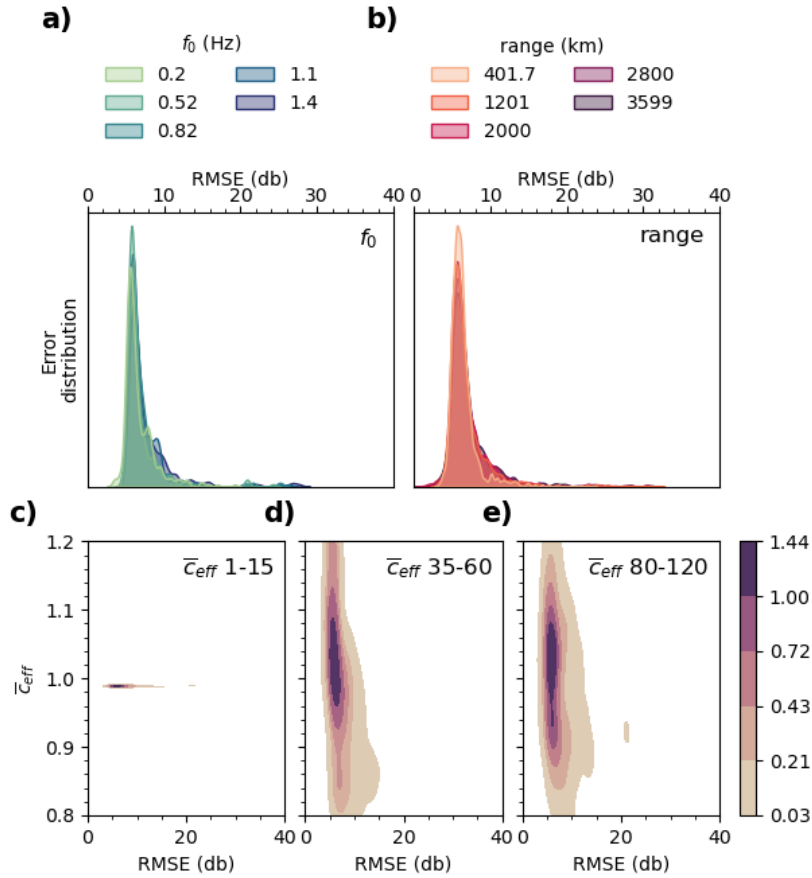
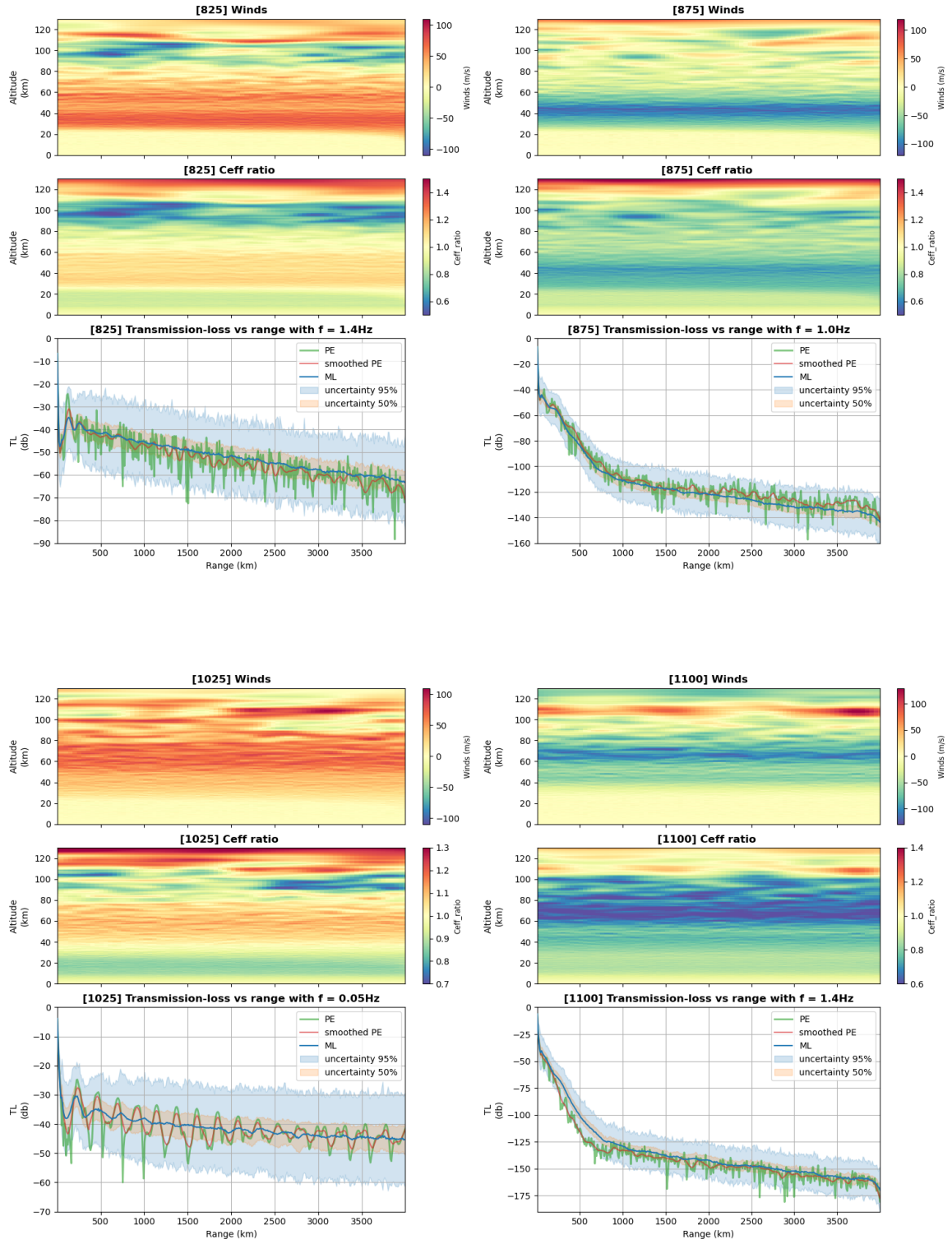


Figure 32: Error distributions for the 1D CNN-RNN with stacked GRUs model (GRU of size 110) over the 4000 km dataset from Alexis Le Pichon. (a) Distribution of RMSE over the testing dataset for various input frequencies. (b) Distribution of RMSE over the testing dataset for various ranges from the source. (c-e) Distribution of RMSE over the testing dataset for various values of effective soundspeed ratio \bar{c}_{eff} in (c) the troposphere, (d) the stratosphere, and (e) the thermosphere.



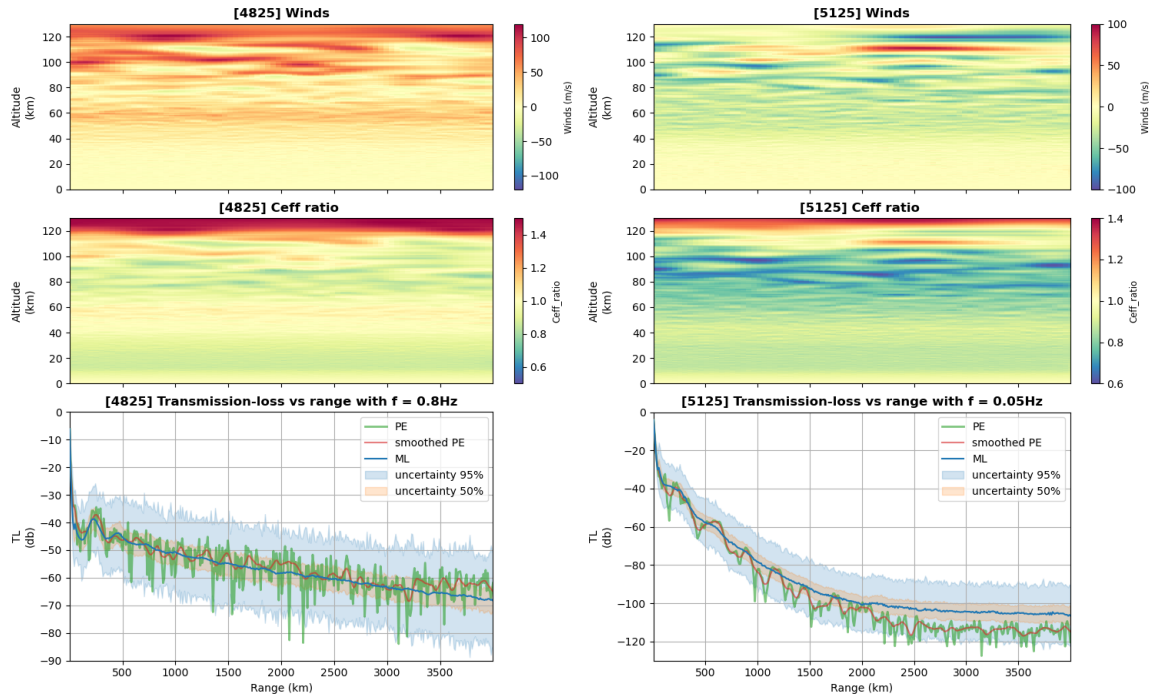


Figure 33: Comparisons for the 1D CNN-RNN with stacked GRUs model (GRU of size 110) over the 4000 km dataset from Alexis Le Pichon between the ML prediction (blue) plotted with its uncertainties (50% confidence and 95% confidence), the original PE results (green) and the smoothed PE results (red). The comparisons are plotted in TL (dB) versus range (km). The wind (m/s) and C_{eff} ratio profiles are shown above (in function of range and altitude in km) to better understand the atmosphere state and visualise the wave guides.

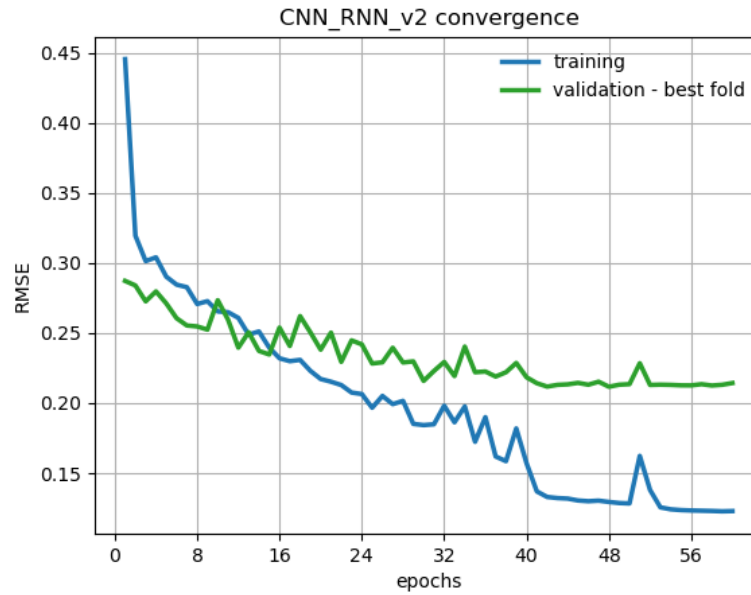


Figure 34: Convergence curves. Evolution of the RMSE with epochs of the 1D CNN-RNN with stacked GRUs model (GRU of size 400) over the training (blue) and testing (green) datasets (4000 km dataset from Alexis Le Pichon).

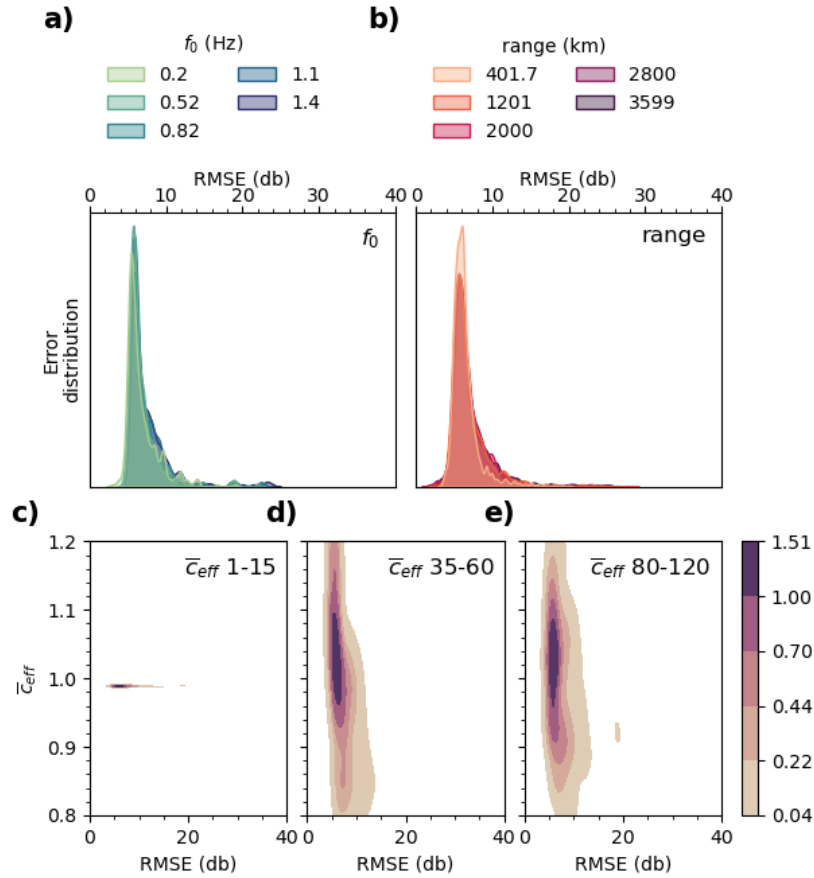
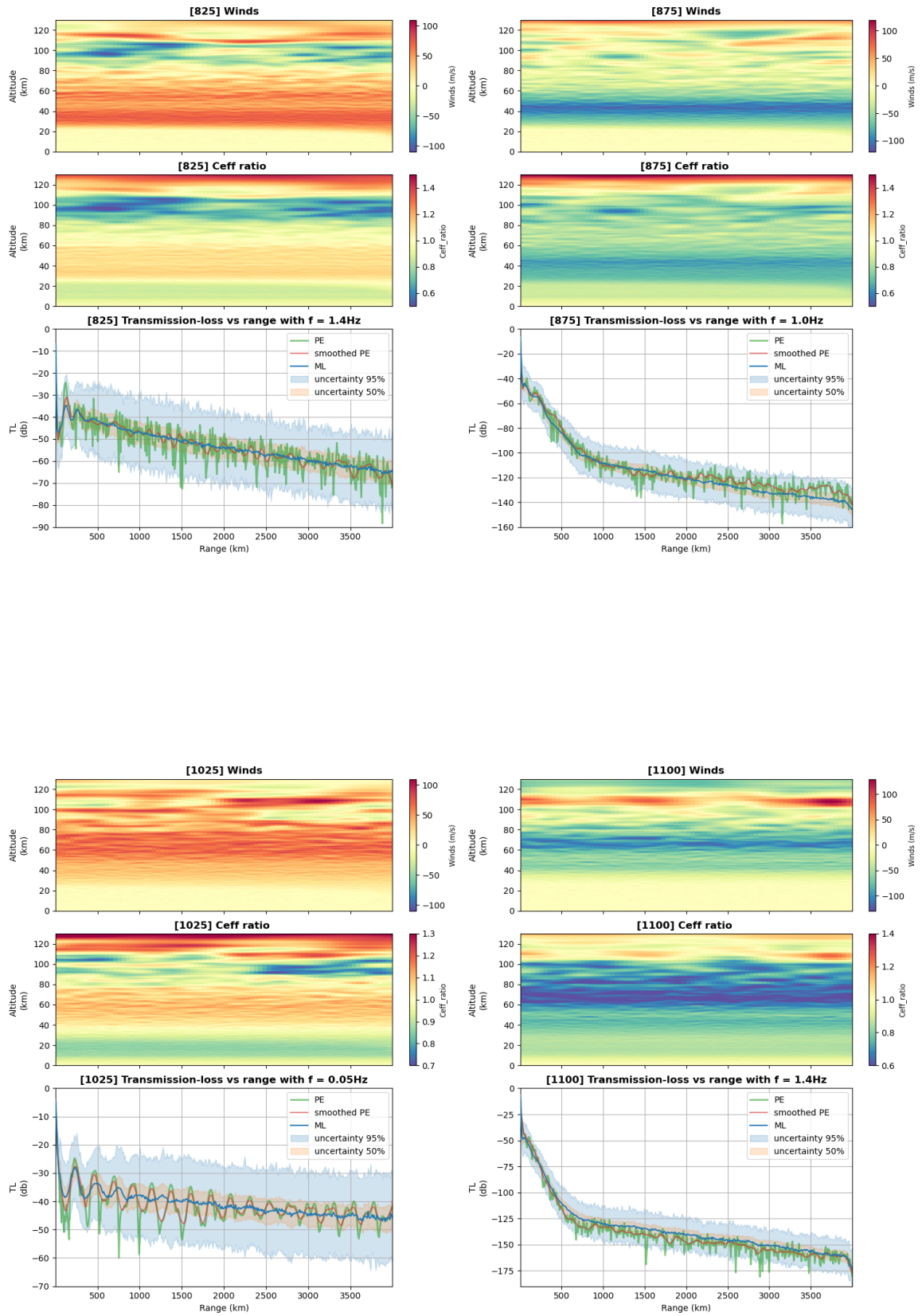


Figure 35: Error distributions for the 1D CNN-RNN with stacked GRUs model (GRU of size 400) over the 4000 km dataset from Alexis Le Pichon. (a) Distribution of RMSE over the testing dataset for various input frequencies. (b) Distribution of RMSE over the testing dataset for various ranges from the source. (c-e) Distribution of RMSE over the testing dataset for various values of effective soundspeed ratio \bar{c}_{eff} in (c) the troposphere, (d) the stratosphere, and (e) the thermosphere.



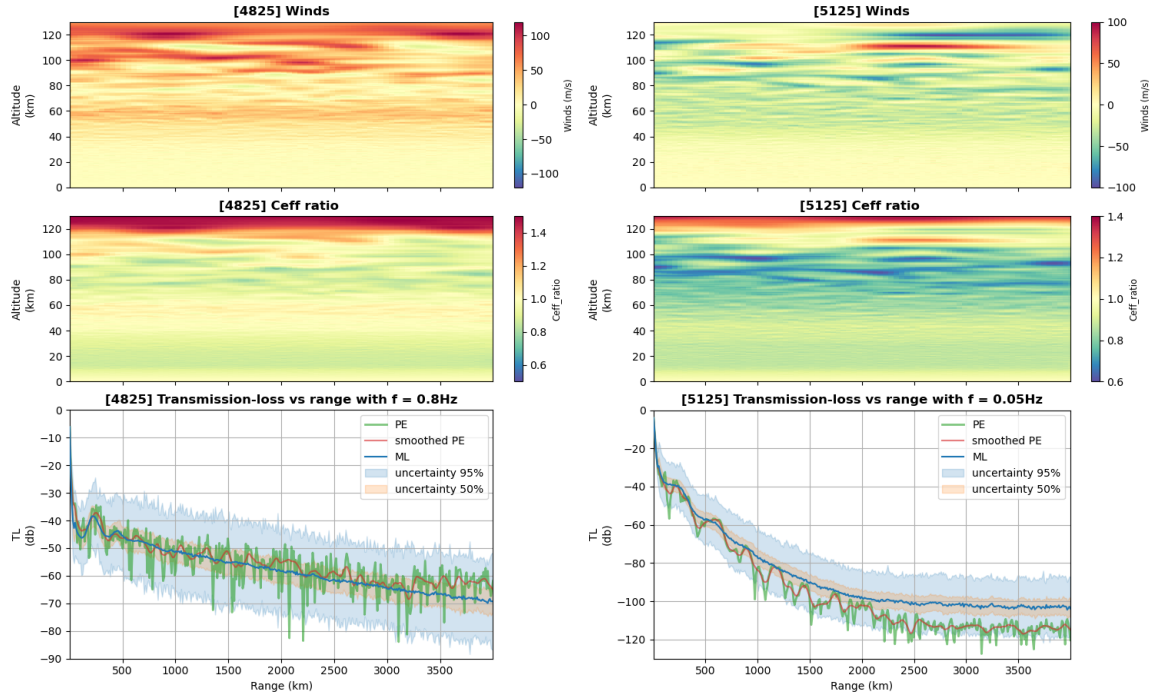


Figure 36: Comparisons for the 1D CNN-RNN with stacked GRUs model (GRU of size 400) over the 4000 km dataset from Alexis Le Pichon between the ML prediction (blue) plotted with its uncertainties (50% confidence and 95% confidence), the original PE results (green) and the smoothed PE results (red). The comparisons are plotted in TL (dB) versus range (km). The wind (m/s) and C_{eff} ratio profiles are shown above (in function of range and altitude in km) to better understand the atmosphere state and visualise the wave guides.

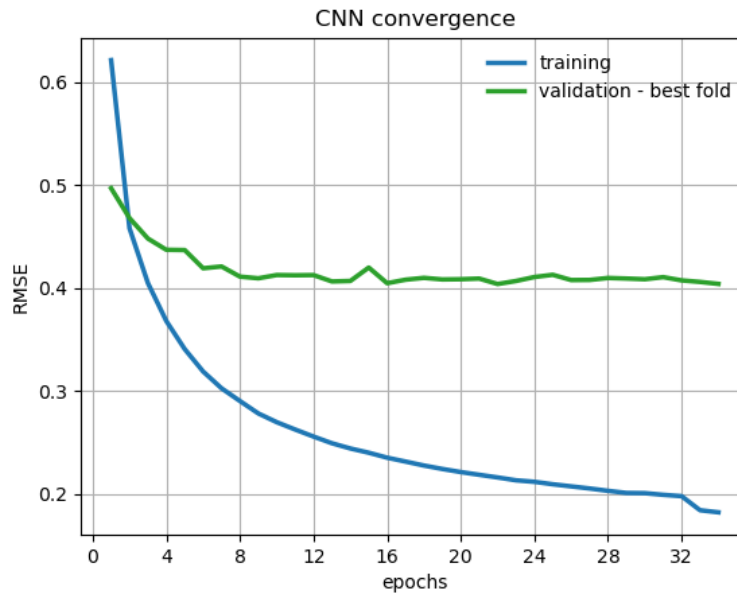


Figure 39: Convergence curves. Evolution of the RMSE with epochs of the 2D CNN model over the training (blue) and testing (green) datasets (new 4000 km dataset).

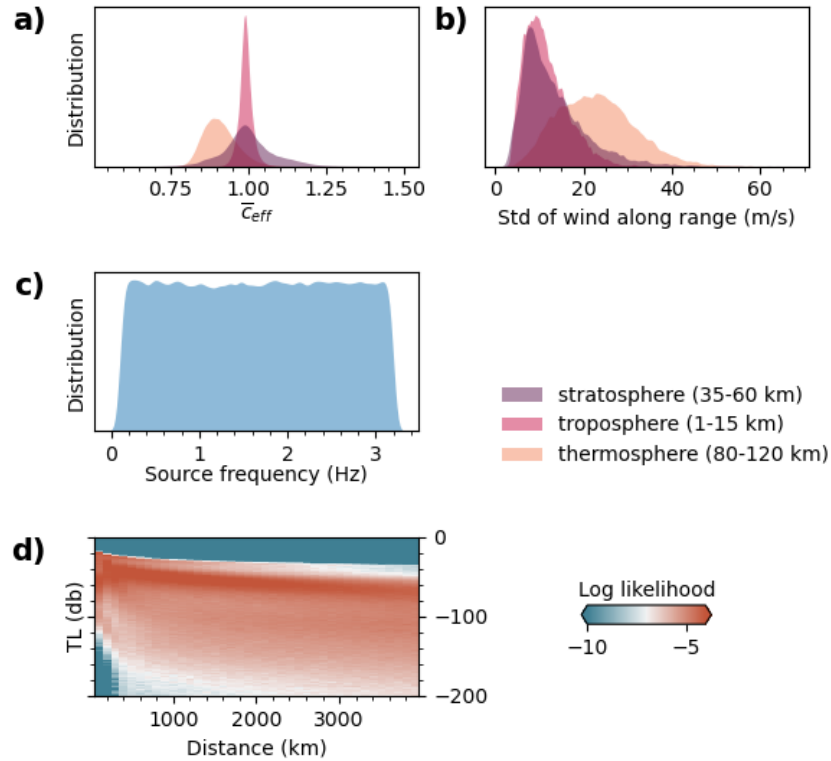


Figure 37: Parameter distribution in the new 4000 km dataset. (a) Distribution of effective soundspeed ratio \bar{c}_{eff} between the ground and various atmospheric layers: troposphere (purple) between 1 and 15 km altitude, troposphere (purple) between 35 and 60 km altitude, and thermosphere (purple) between 80 and 120 km altitude. (b) Distribution of standard deviations of wind velocities along range for various atmospheric layers. (c) Distribution of input source frequencies used in PE simulations to build the entire TL dataset. (d) TL distribution represented as log likelihood (computed from Gaussian Kernel density estimates) vs distance determined from our entire TL dataset.

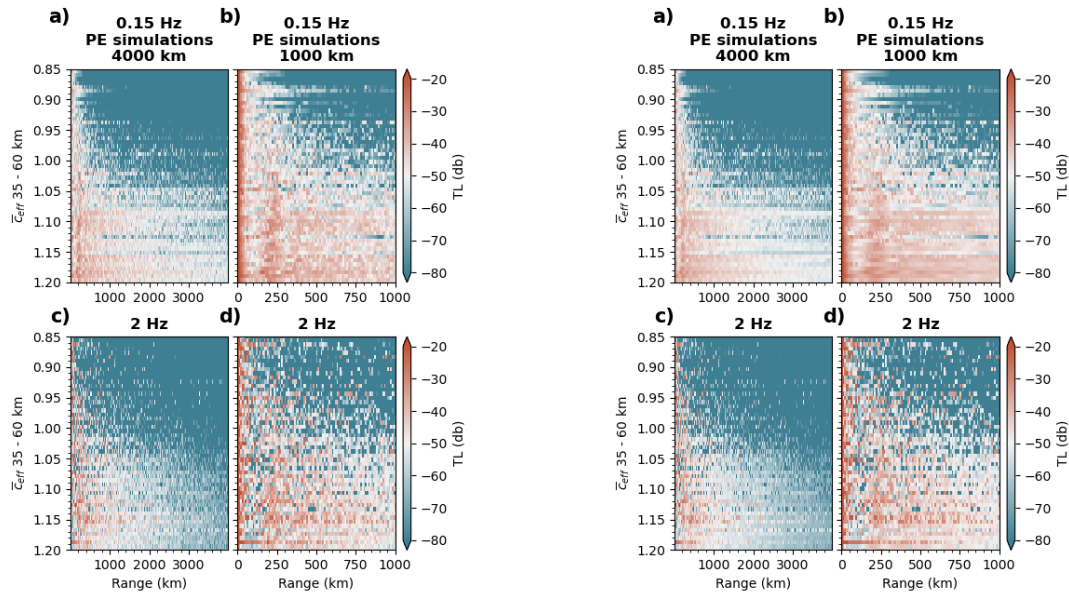


Figure 38: Original (left) and smoothed (right) TL map

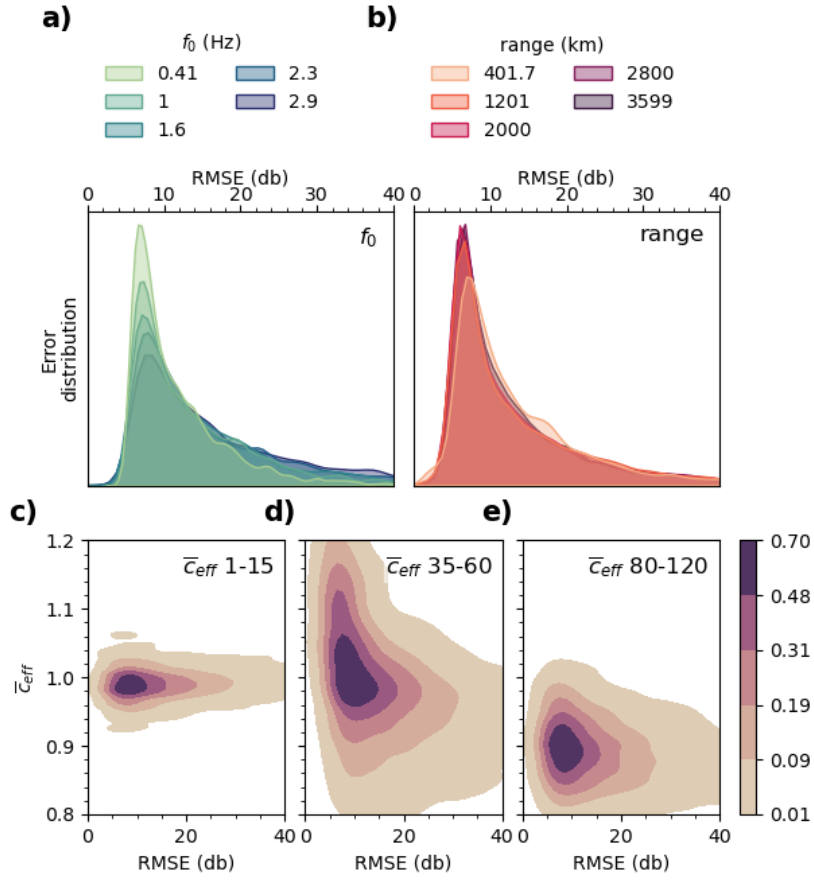
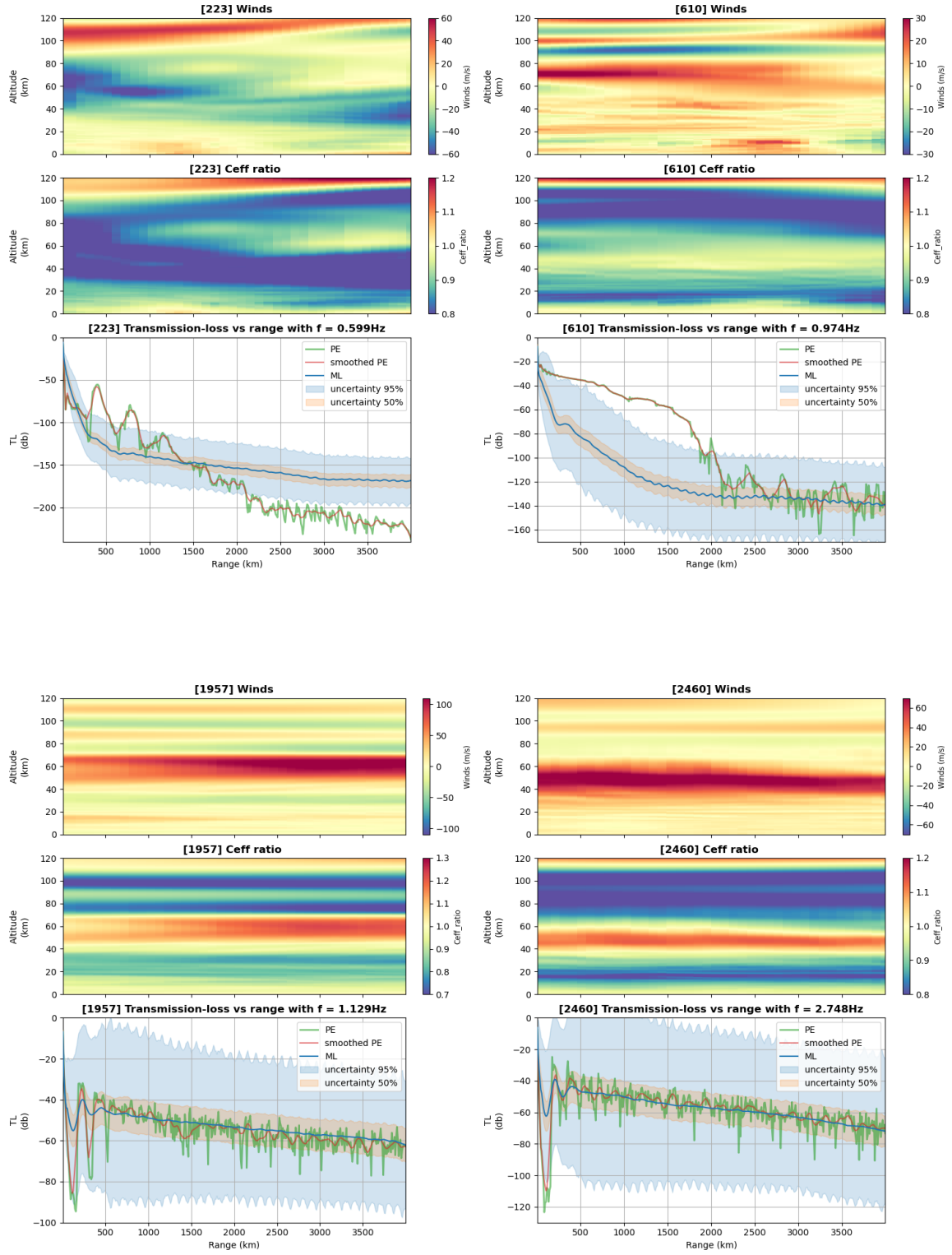


Figure 40: Error distributions for the 2D CNN model over the new 4000 km dataset. (a) Distribution of RMSE over the testing dataset for various input frequencies. (b) Distribution of RMSE over the testing dataset for various ranges from the source. (c-e) Distribution of RMSE over the testing dataset for various values of effective sound speed ratio \bar{c}_{eff} in (c) the troposphere, (d) the stratosphere, and (e) the thermosphere.



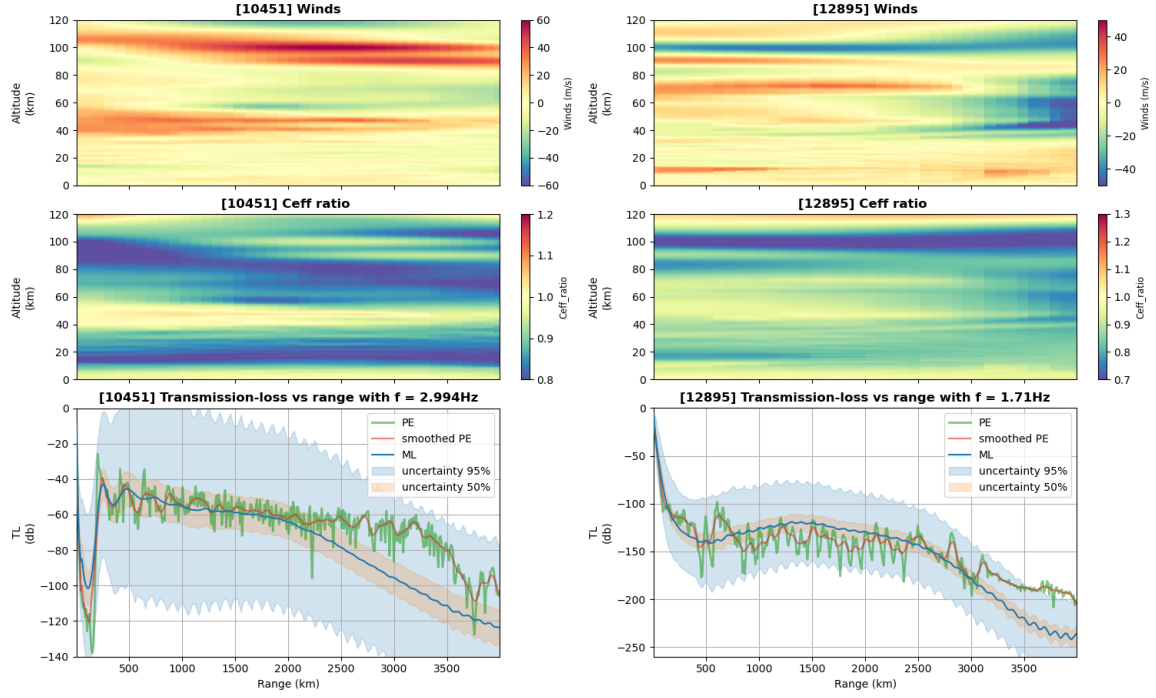


Figure 41: Comparisons for the 2D CNN model over the new 4000 km dataset between the ML prediction (blue) plotted with its uncertainties (50% confidence and 95% confidence), the original PE results (green) and the smoothed PE results (red). The comparisons are plotted in TL (dB) versus range (km). The wind (m/s) and C_{eff} ratio profiles are shown above (in function of range and altitude in km) to better understand the atmosphere state and visualise the wave guides.

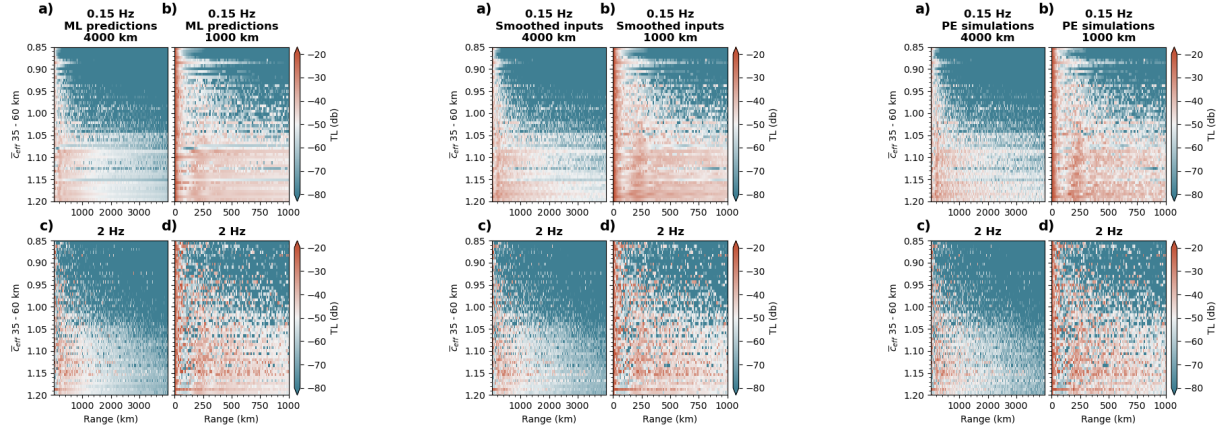


Figure 42: TL predictions (in dB) for the 2D CNN model over the new 4000 km dataset vs distance from the source (in km) and C_{eff} for the ML model (left), smoothed NCPA-simulated TL (middle), full NCPA-simulated TL (right) at 0.15 Hz (panels a and b) and 2 Hz (panels c and d).

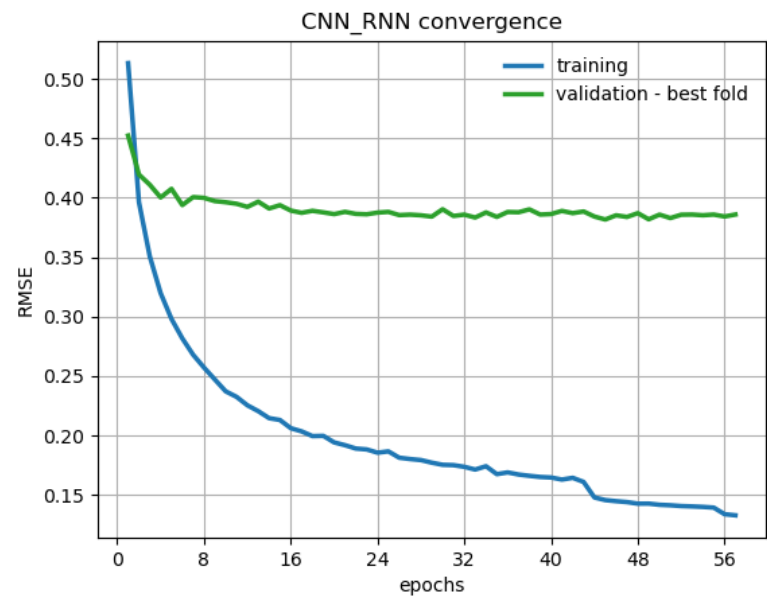


Figure 43: Convergence curves. Evolution of the RMSE with epochs of the 1D CNN-RNN model over the training (blue) and testing (green) datasets (new 4000 km dataset).

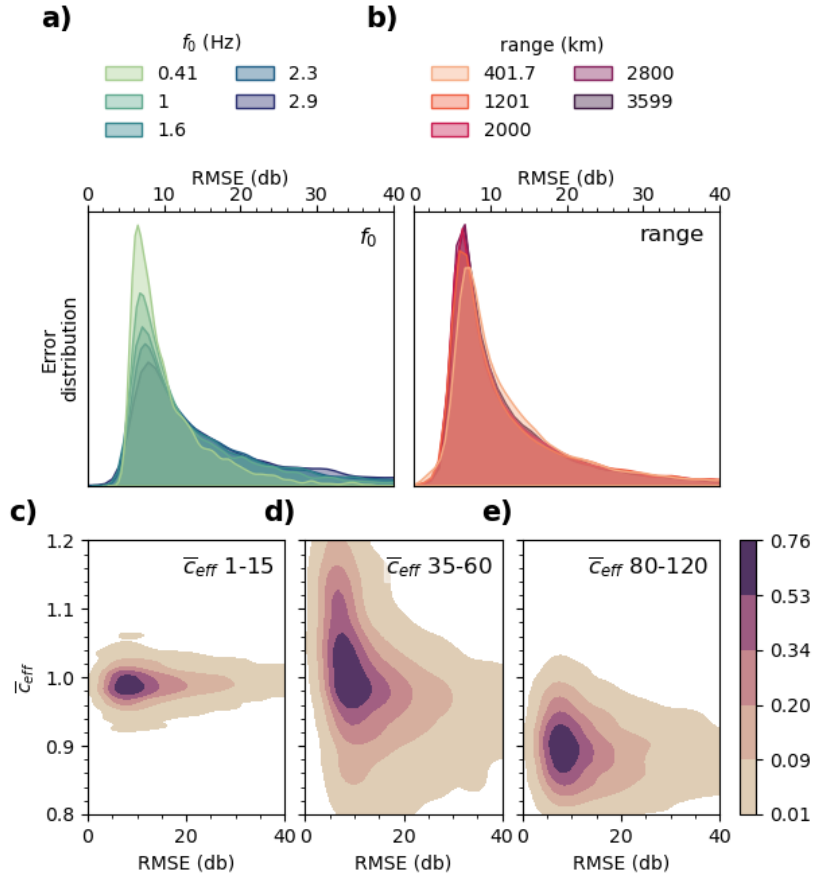
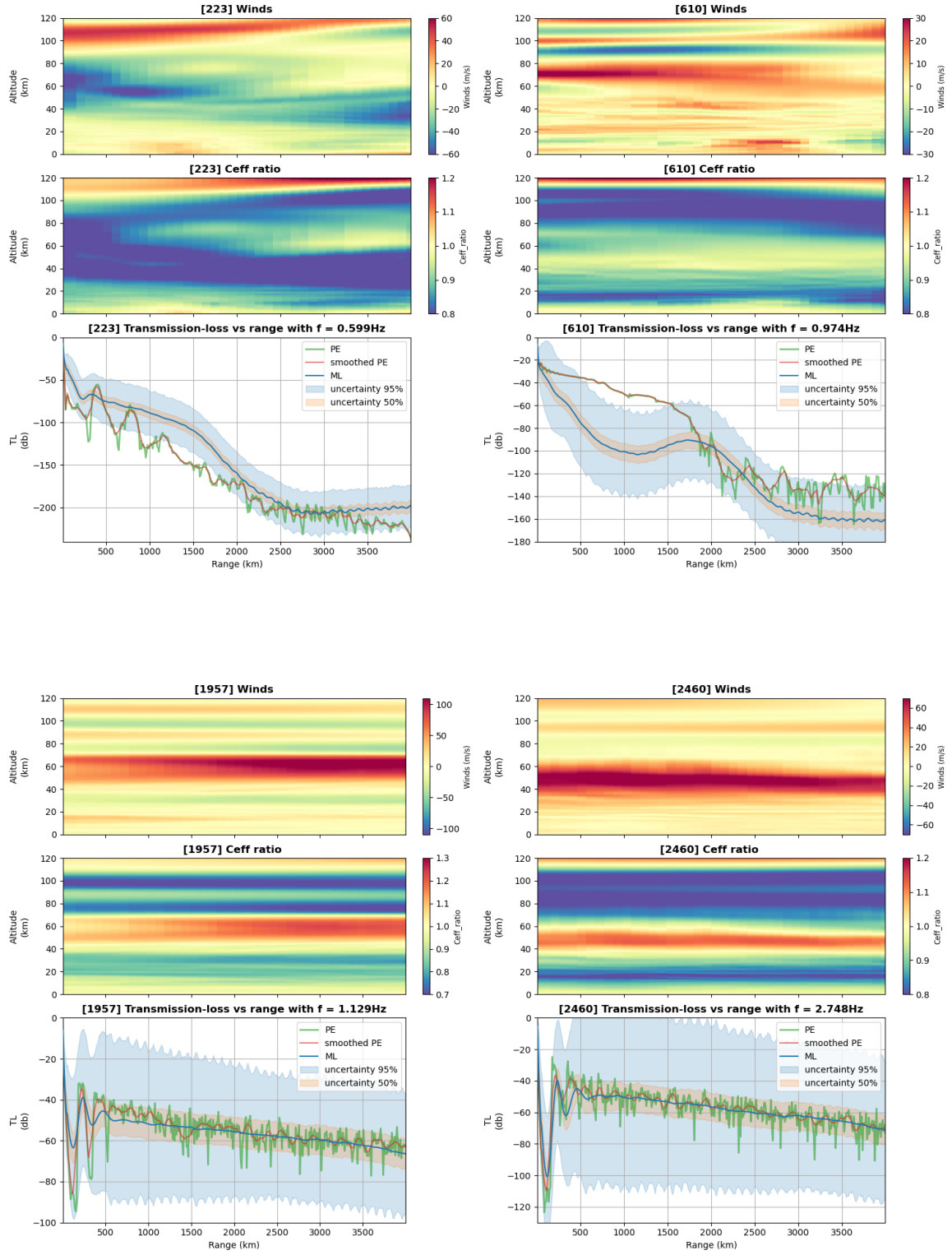


Figure 44: Error distributions for the 1D CNN-RNN model over the new 4000 km dataset. (a) Distribution of RMSE over the testing dataset for various input frequencies. (b) Distribution of RMSE over the testing dataset for various ranges from the source. (c-e) Distribution of RMSE over the testing dataset for various values of effective soundspeed ratio \bar{c}_{eff} in (c) the troposphere, (d) the stratosphere, and (e) the thermosphere.



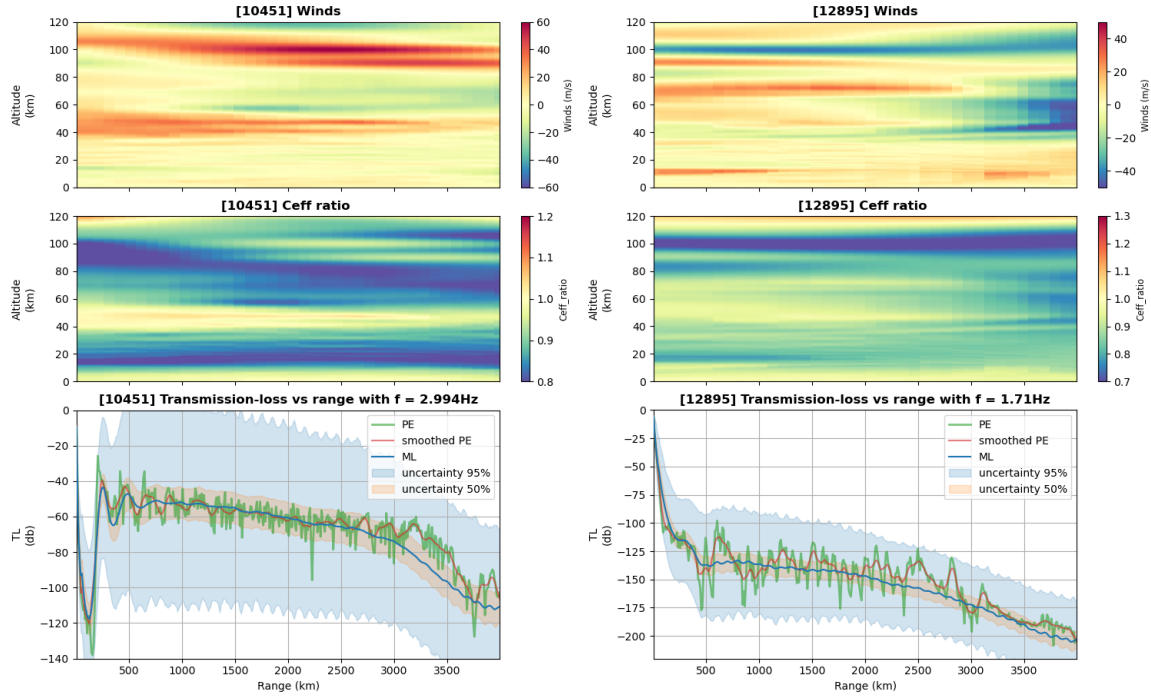


Figure 45: Comparisons for the 1D CNN-RNN model over the new 4000 km dataset between the ML prediction (blue) plotted with its uncertainties (50% confidence and 95% confidence), the original PE results (green) and the smoothed PE results (red). The comparisons are plotted in TL (dB) versus range (km). The wind (m/s) and C_{eff} ratio profiles are shown above (in function of range and altitude in km) to better understand the atmosphere state and visualise the wave guides.

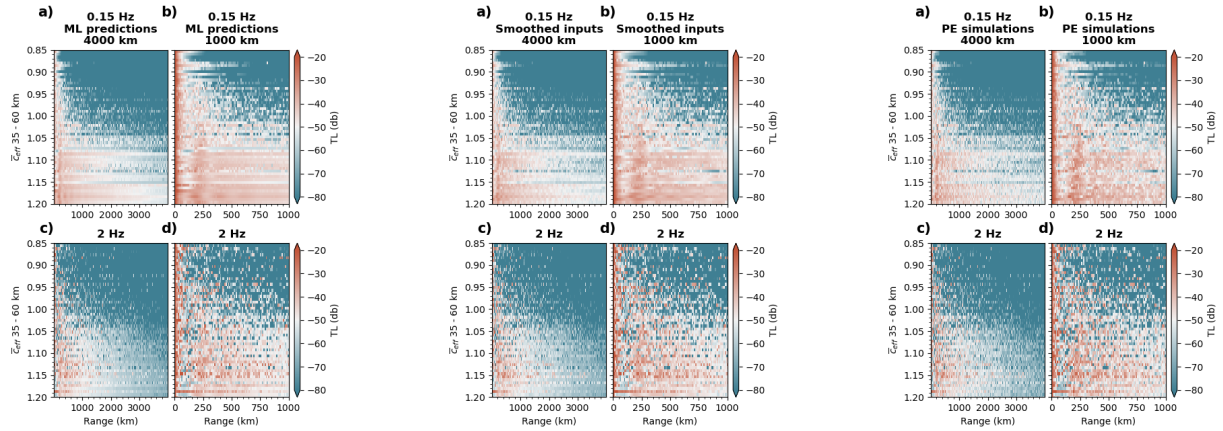


Figure 46: TL predictions (in db) for the 1D CNN-RNN model over the new 4000 km dataset vs distance from the source (in km) and C_{eff} for the ML model (left), smoothed NCPA-simulated TL (middle), full NCPA-simulated TL (right) at 0.15 Hz (panels a and b) and 2 Hz (panels c and d).

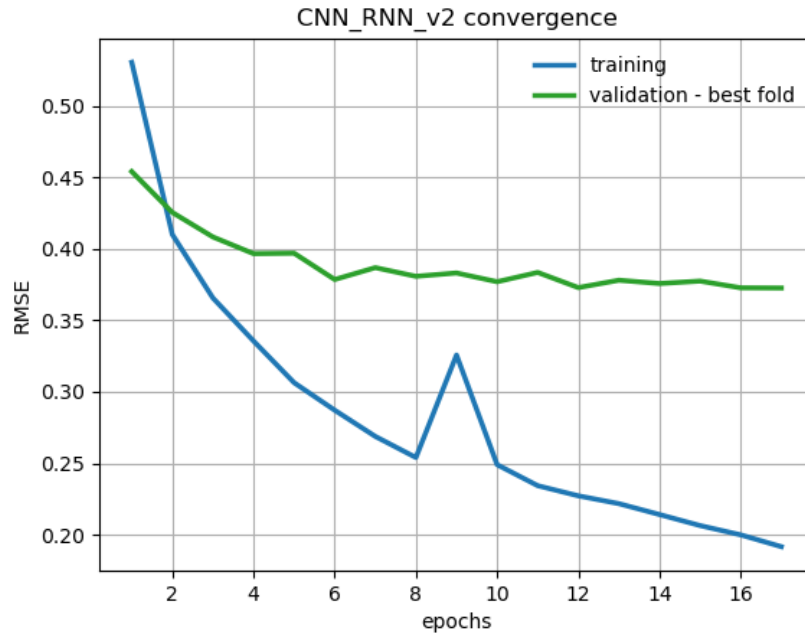
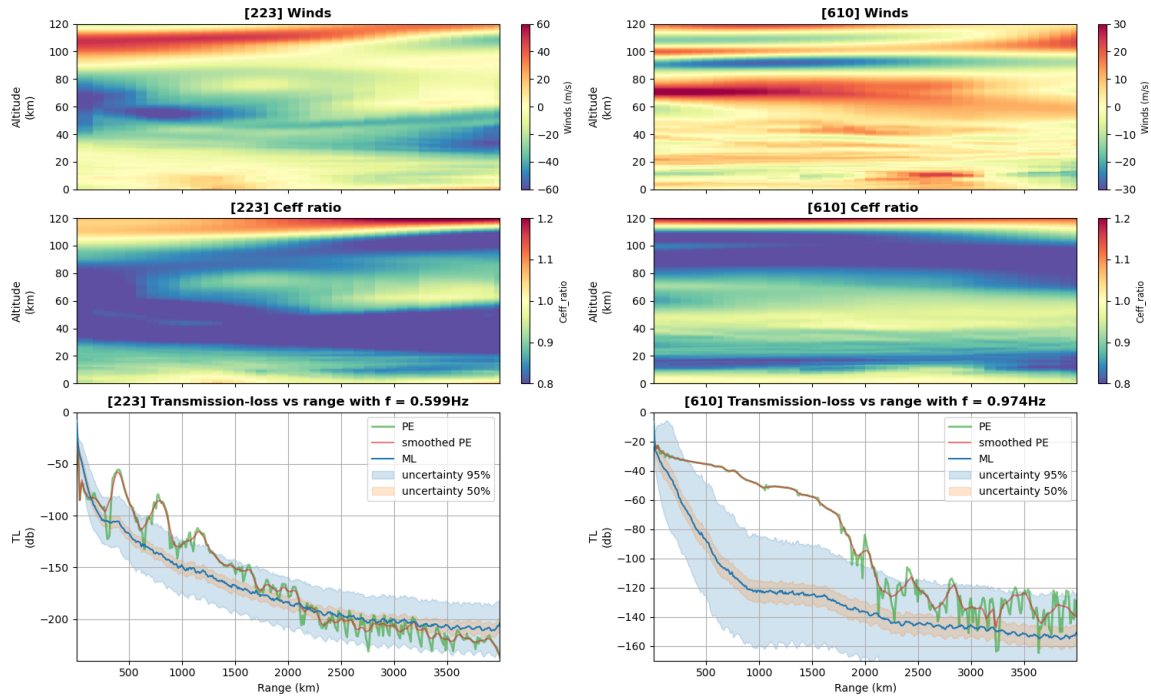


Figure 47: Convergence curves. Evolution of the RMSE with epochs of the 1D CNN-RNN with stacked GRUs model (GRU of size 400) over the training (blue) and testing (green) datasets (new 4000 km dataset).



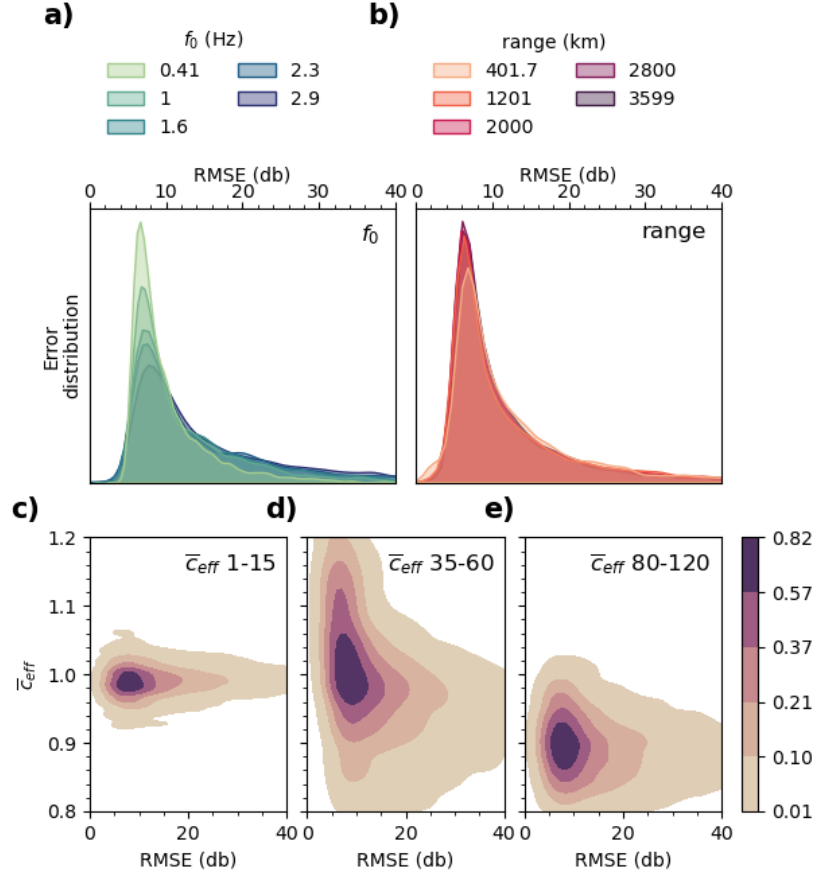


Figure 48: Error distributions for the 1D CNN-RNN with stacked GRUs model (GRU of size 400) over the new 4000 km dataset. (a) Distribution of RMSE over the testing dataset for various input frequencies. (b) Distribution of RMSE over the testing dataset for various ranges from the source. (c-e) Distribution of RMSE over the testing dataset for various values of effective soundspeed ratio \bar{c}_{eff} in (c) the troposphere, (d) the stratosphere, and (e) the thermosphere.

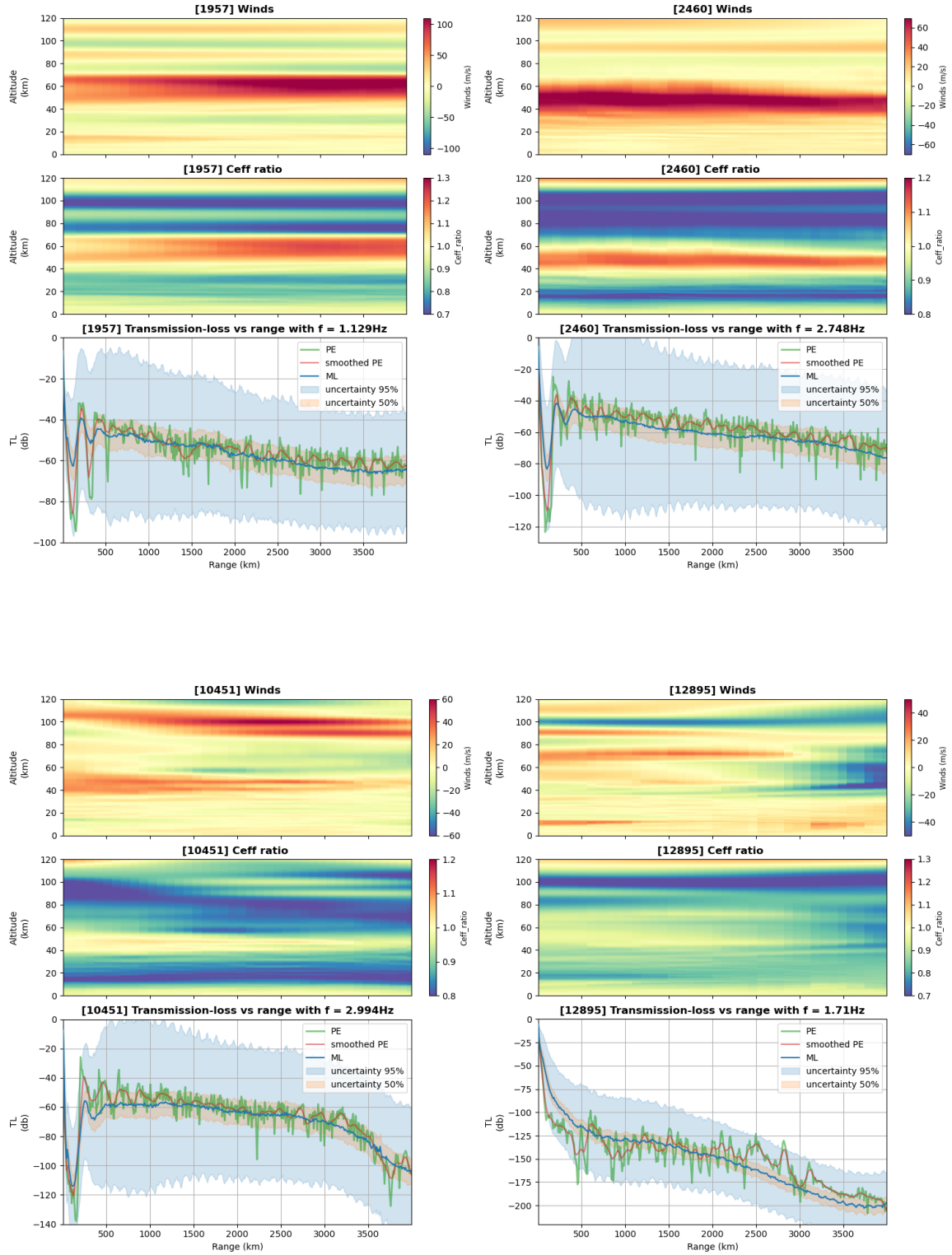


Figure 49: Comparisons for the 1D CNN-RNN with stacked GRUs model (GRU of size 400) over the new 4000 km dataset between the ML prediction (blue) plotted with its uncertainties (50% confidence and 95% confidence), the original PE results (green) and the smoothed PE results (red). The comparisons are plotted in TL (dB) versus range (km). The wind (m/s) and C_{eff} ratio profiles are shown above (in function of range and altitude in km) to better understand the atmosphere state and visualise the wave guides.

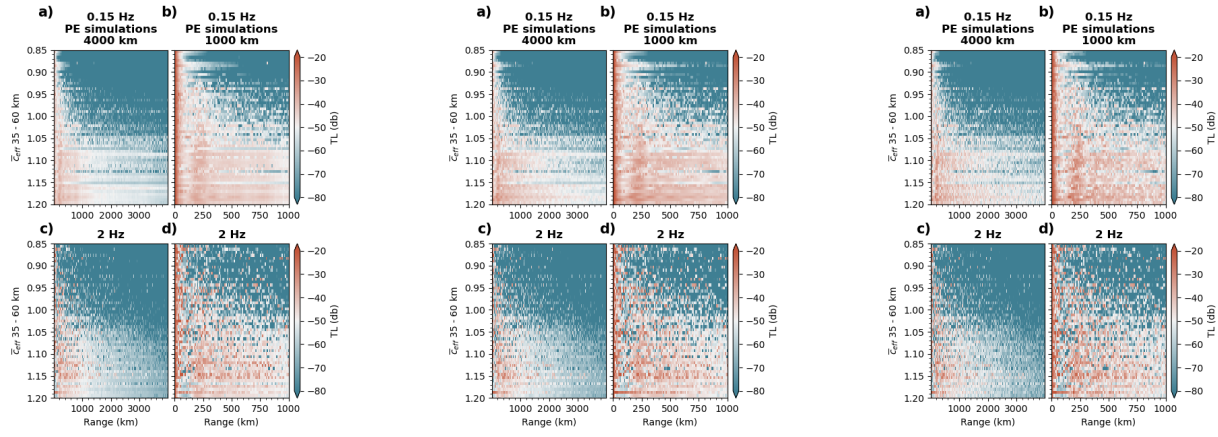


Figure 50: TL predictions (in db) for the 1D CNN-RNN with stacked GRUs model (GRU of size 400) over the new 4000 km dataset vs distance from the source (in km) and C_{eff} for the ML model (left), smoothed NCPA-simulated TL (middle), full NCPA-simulated TL (right) at 0.15 Hz (panels a and b) and 2 Hz (panels c and d).

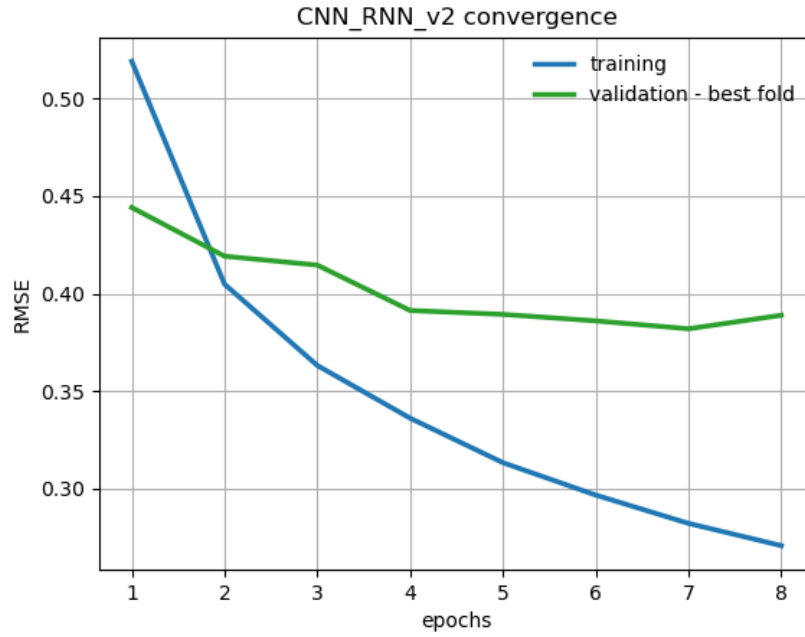


Figure 51: Convergence curves. Evolution of the RMSE with epochs of the 1D CNN-RNN with stacked GRUs model (GRU of size 110) over the training (blue) and testing (green) datasets (new 4000 km dataset).

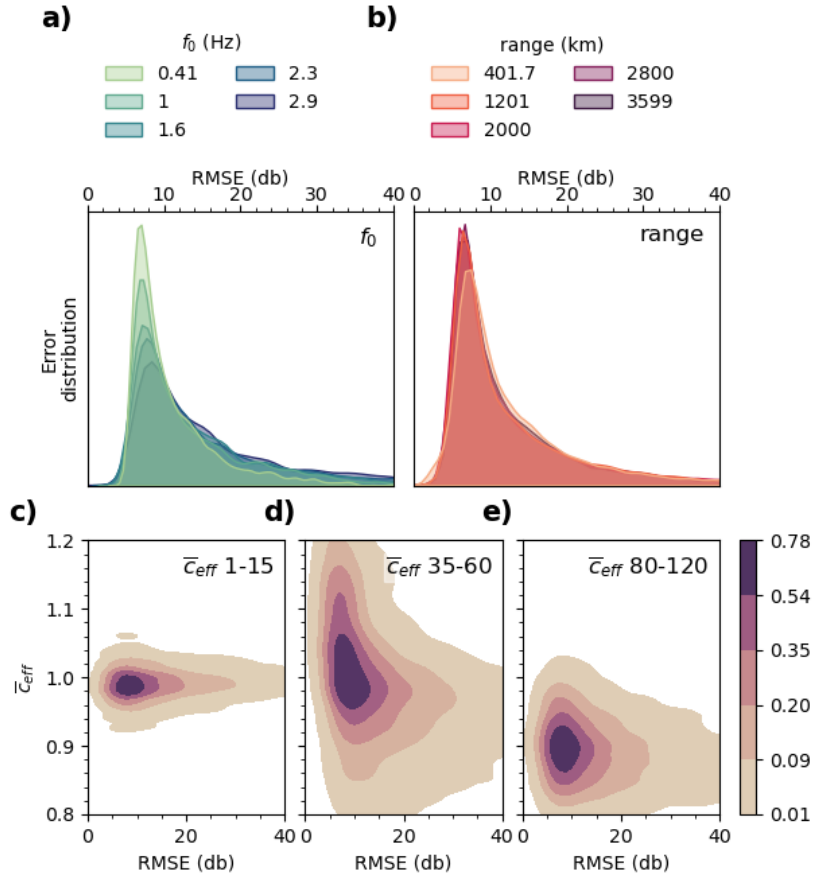
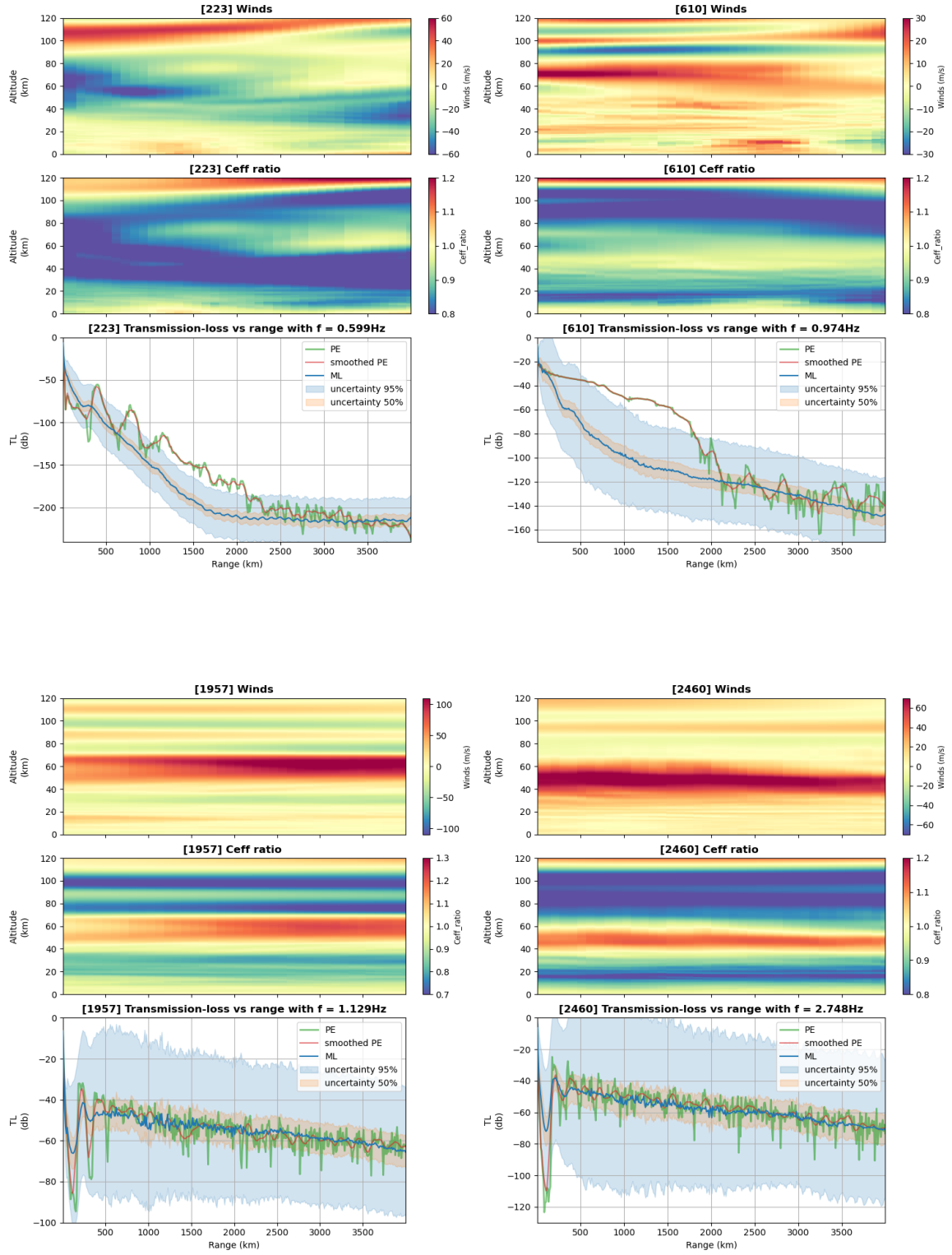


Figure 52: Error distributions for the 1D CNN-RNN with stacked GRUs model (GRU of size 110) over the new 4000 km dataset. (a) Distribution of RMSE over the testing dataset for various input frequencies. (b) Distribution of RMSE over the testing dataset for various ranges from the source. (c-e) Distribution of RMSE over the testing dataset for various values of effective soundspeed ratio \bar{c}_{eff} in (c) the troposphere, (d) the stratosphere, and (e) the thermosphere.



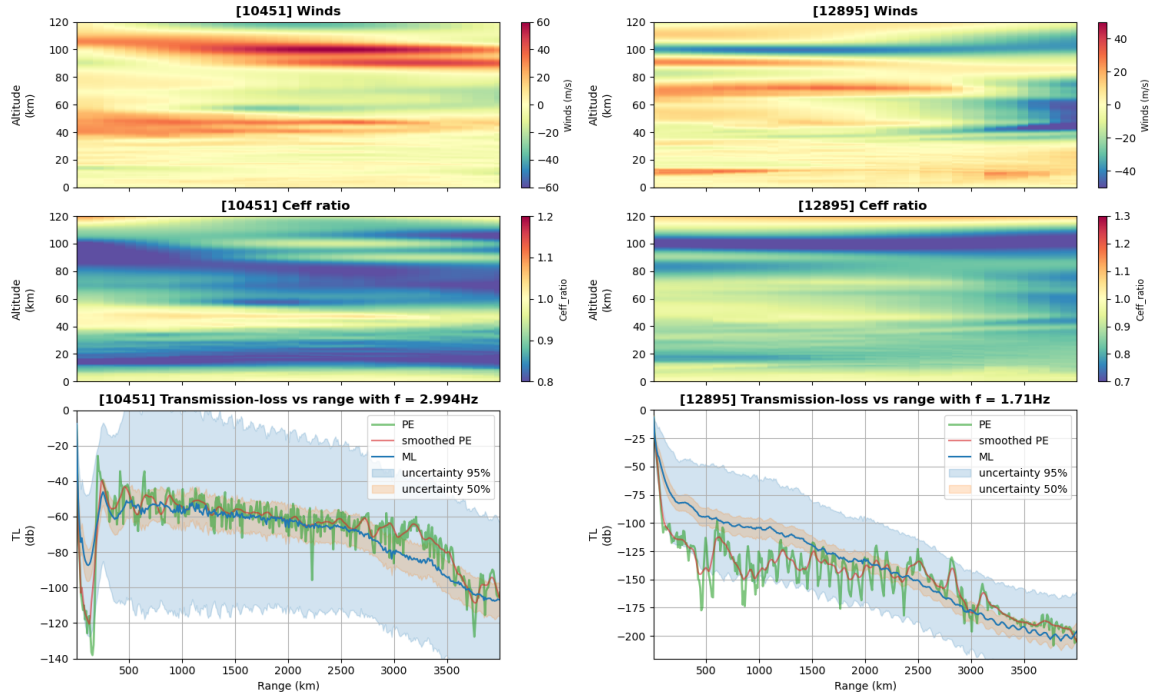


Figure 53: Comparisons for the 1D CNN-RNN with stacked GRUs model (GRU of size 110) over the new 4000 km dataset between the ML prediction (blue) plotted with its uncertainties (50% confidence and 95% confidence), the original PE results (green) and the smoothed PE results (red). The comparisons are plotted in TL (dB) versus range (km). The wind (m/s) and C_{eff} ratio profiles are shown above (in function of range and altitude in km) to better understand the atmosphere state and visualise the wave guides.

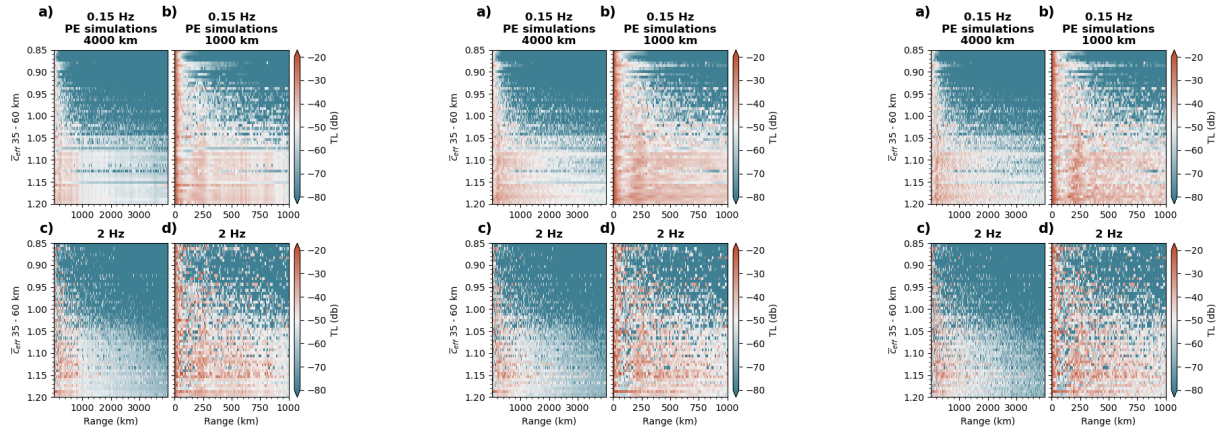


Figure 54: TL predictions (in db) for the 1D CNN-RNN with stacked GRUs model (GRU of size 110) over the new 4000 km dataset vs distance from the source (in km) and C_{eff} for the ML model (left), smoothed NCPA-simulated TL (middle), full NCPA-simulated TL (right) at 0.15 Hz (panels a and b) and 2 Hz (panels c and d).

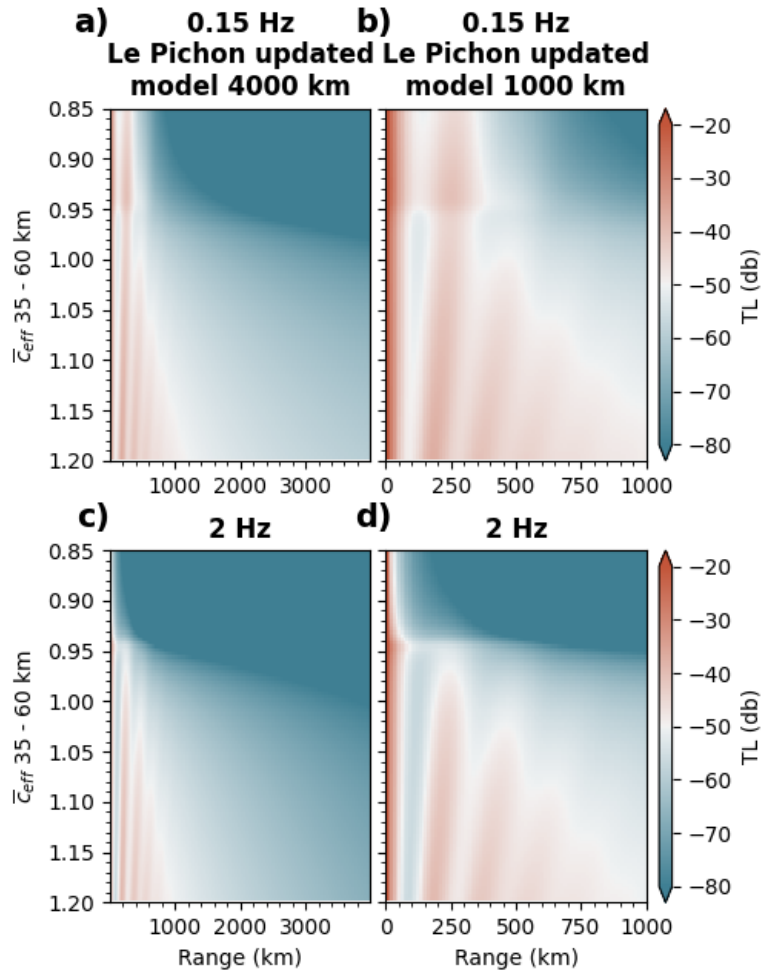


Figure 55: Map of TL values (in dB) based on the updated attenuation law, in function of C_{eff} and range for two different frequencies (0.15 Hz in panels a and b, 2 Hz for panels c and d). a and c panels are for 0-4000km and b) and d) for 0-1000km

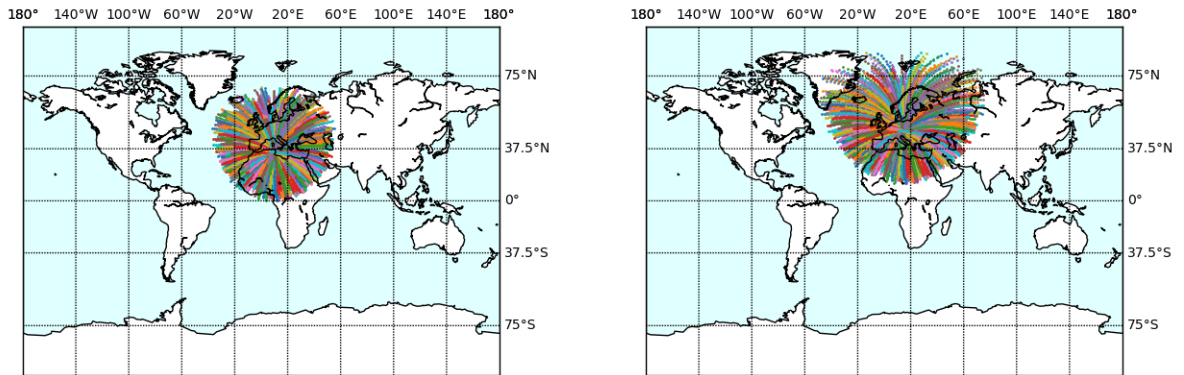


Figure 56: Slices crossing I48TN (longitude 9.32302°, latitude 35.80523°) at various ranges and angles (left), and slices crossing I26DE (longitude 13.712018°, latitude 48.844365°) at various ranges and angles (right)

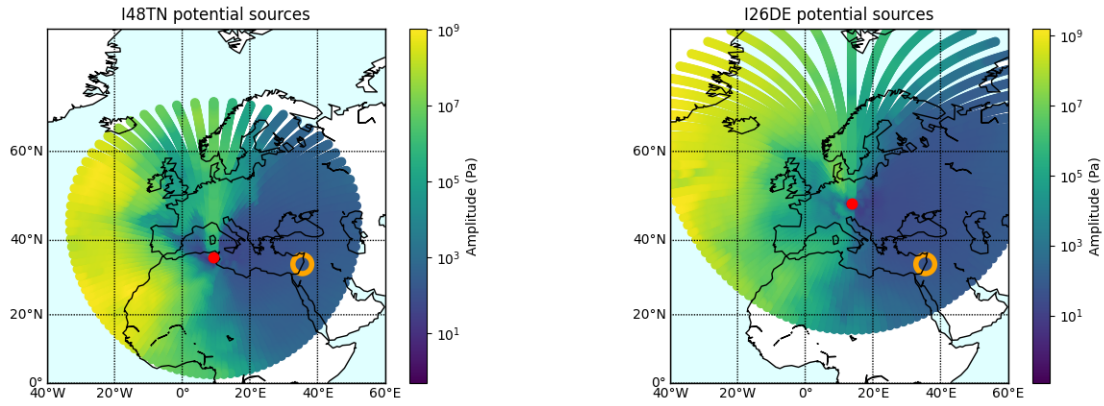


Figure 57: Interpolated pressure map in Pa around I48TN (longitude 9.32302° , latitude 35.80523°) at various ranges and angles (left), and same around I48TN I26DE (longitude 13.712018° , latitude 48.844365°) (right). The red dots are the stations' locations (I48TN in Tunisia and I26DE in Germany), and the orange circle is Beirut's location (longitude 35.50° , latitude 33.89°)

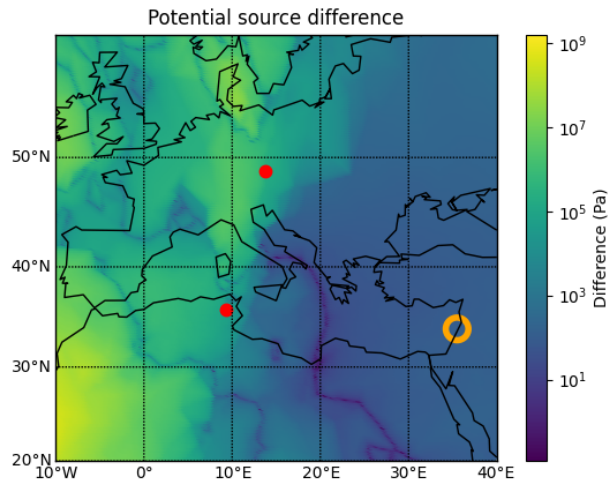


Figure 58: Difference of the interpolated pressure maps in Pa. The red dots are the stations' locations (I48TN in Tunisia and I26DE in Germany), and the orange circle is Beirut's location (longitude 35.50° , latitude 33.89°).

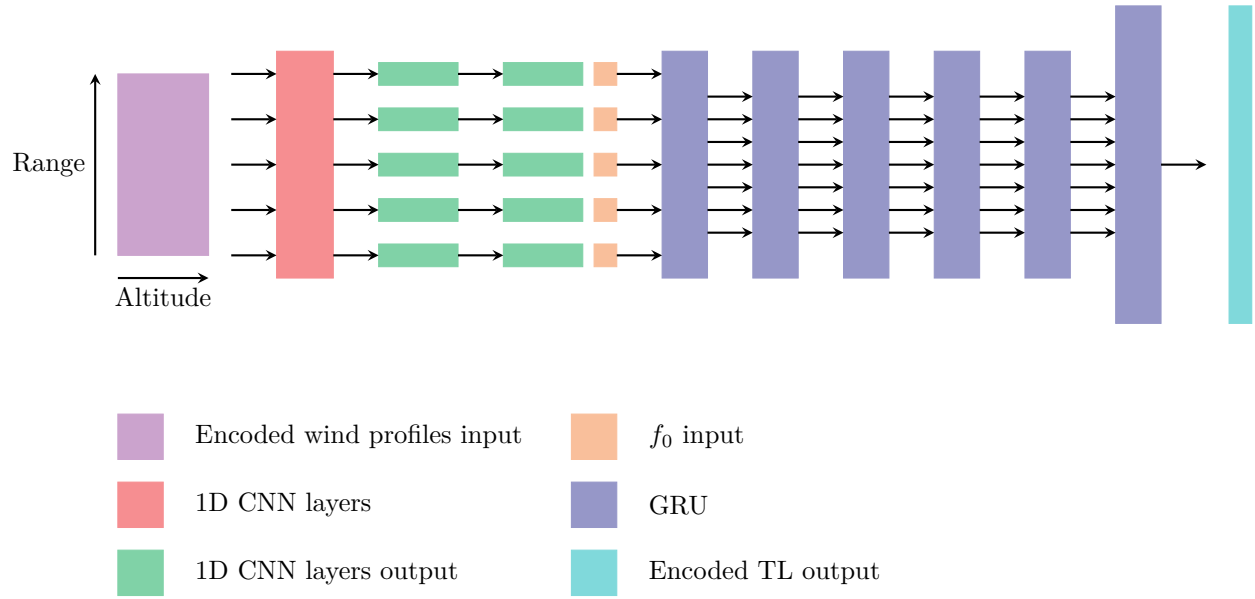
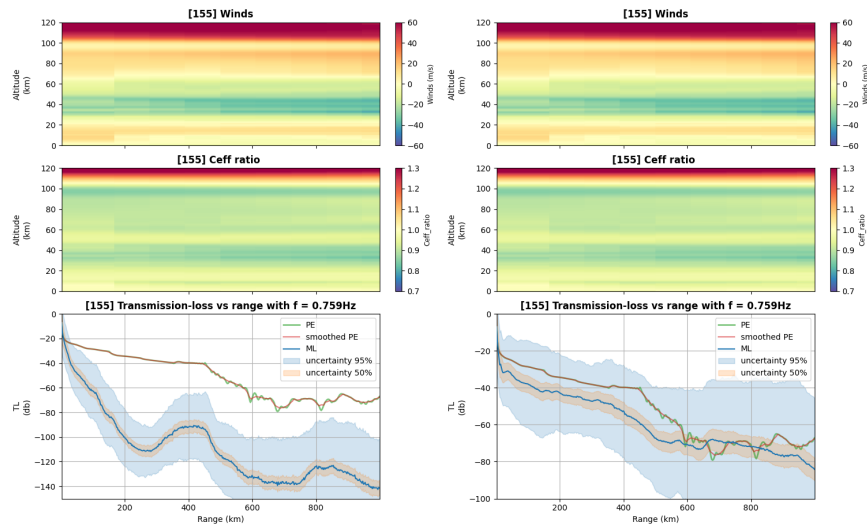


Figure 59: CNN and stacked GRUs architecture



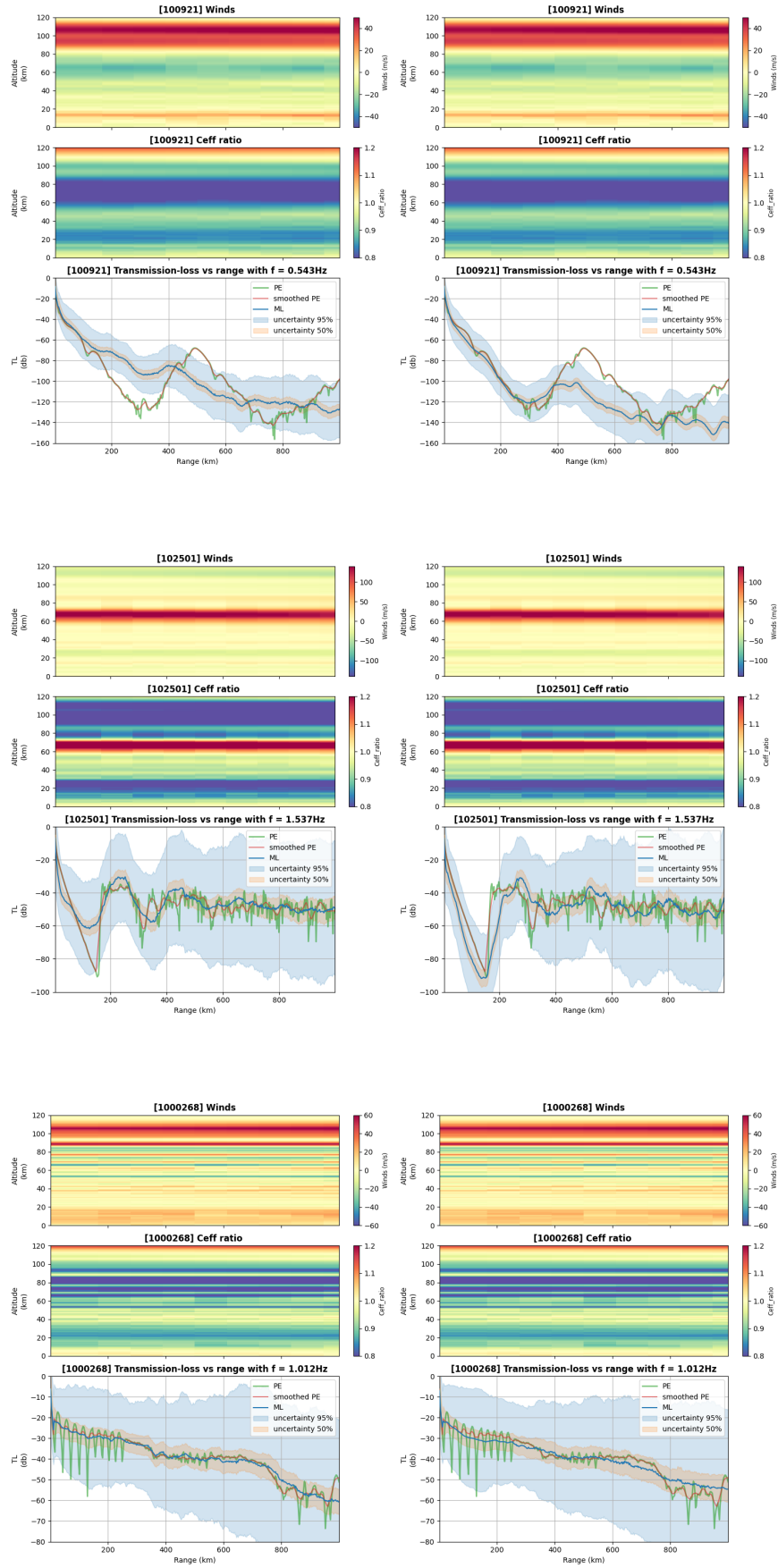
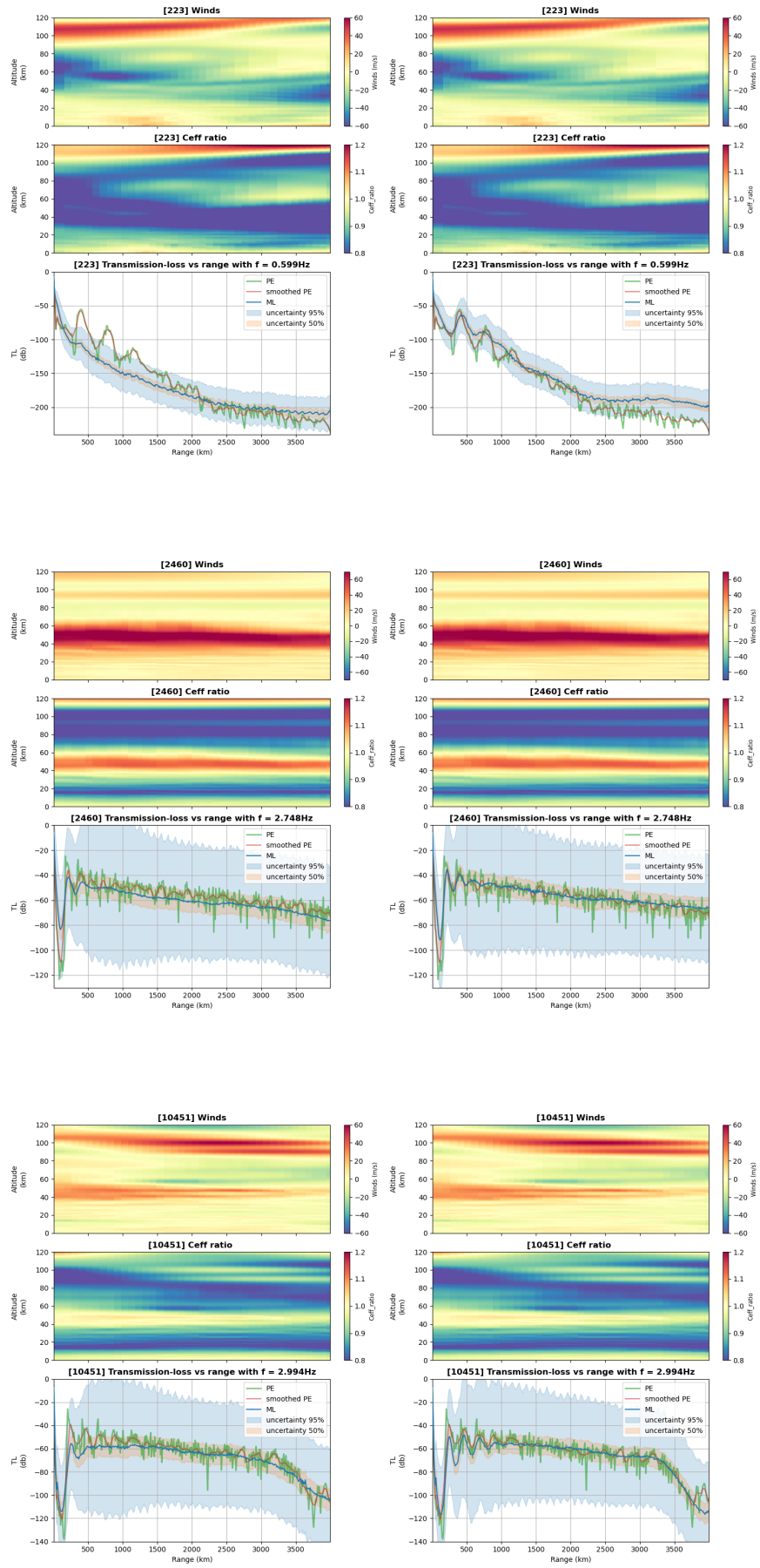


Figure 60: Left: 3 GRU units, right: 6 GRU units



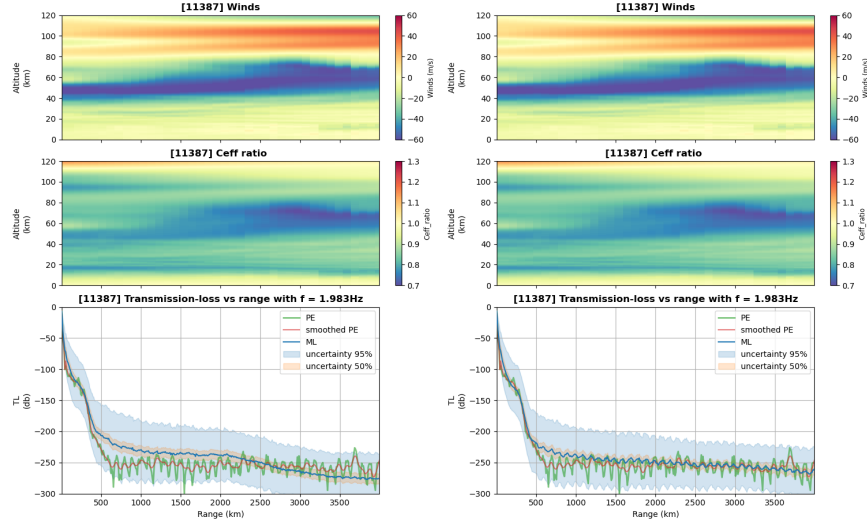


Figure 61: Left: 3 GRU units, right: 6 GRU units

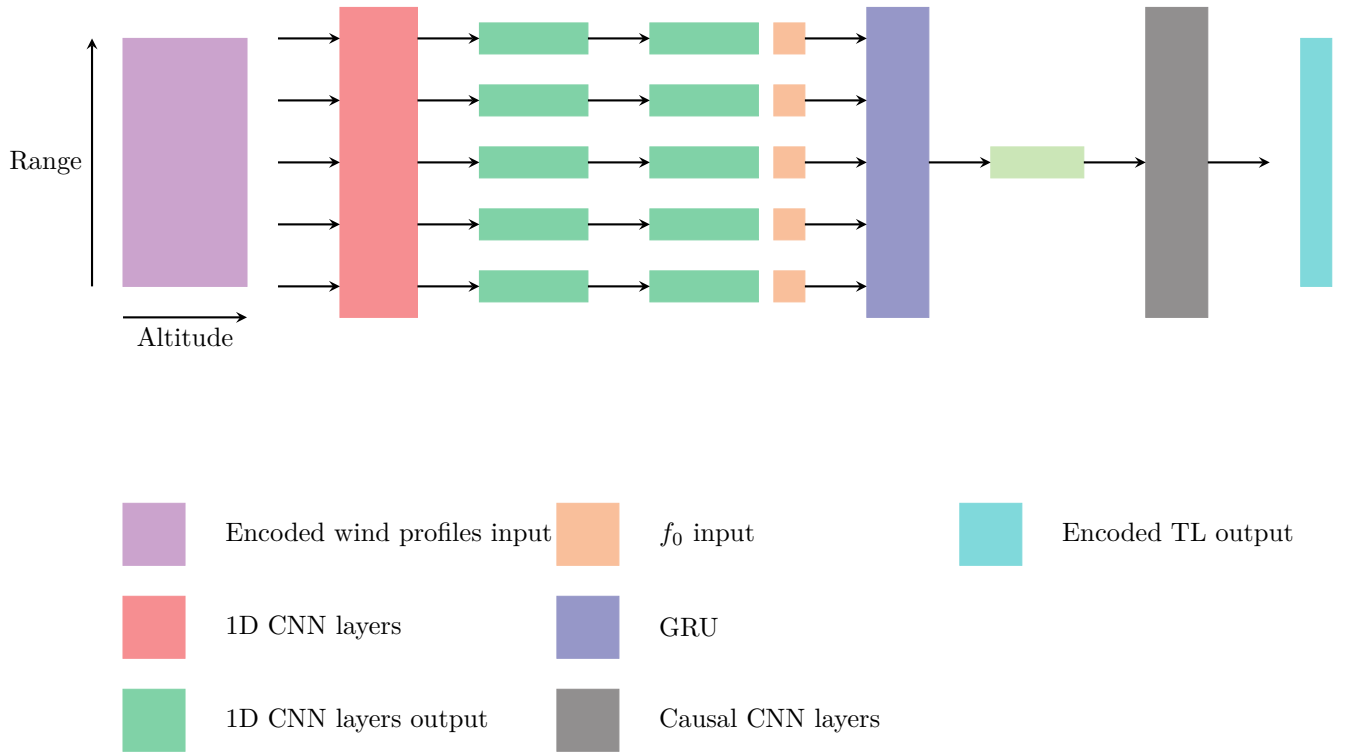


Figure 62: 1D CNN and GRU

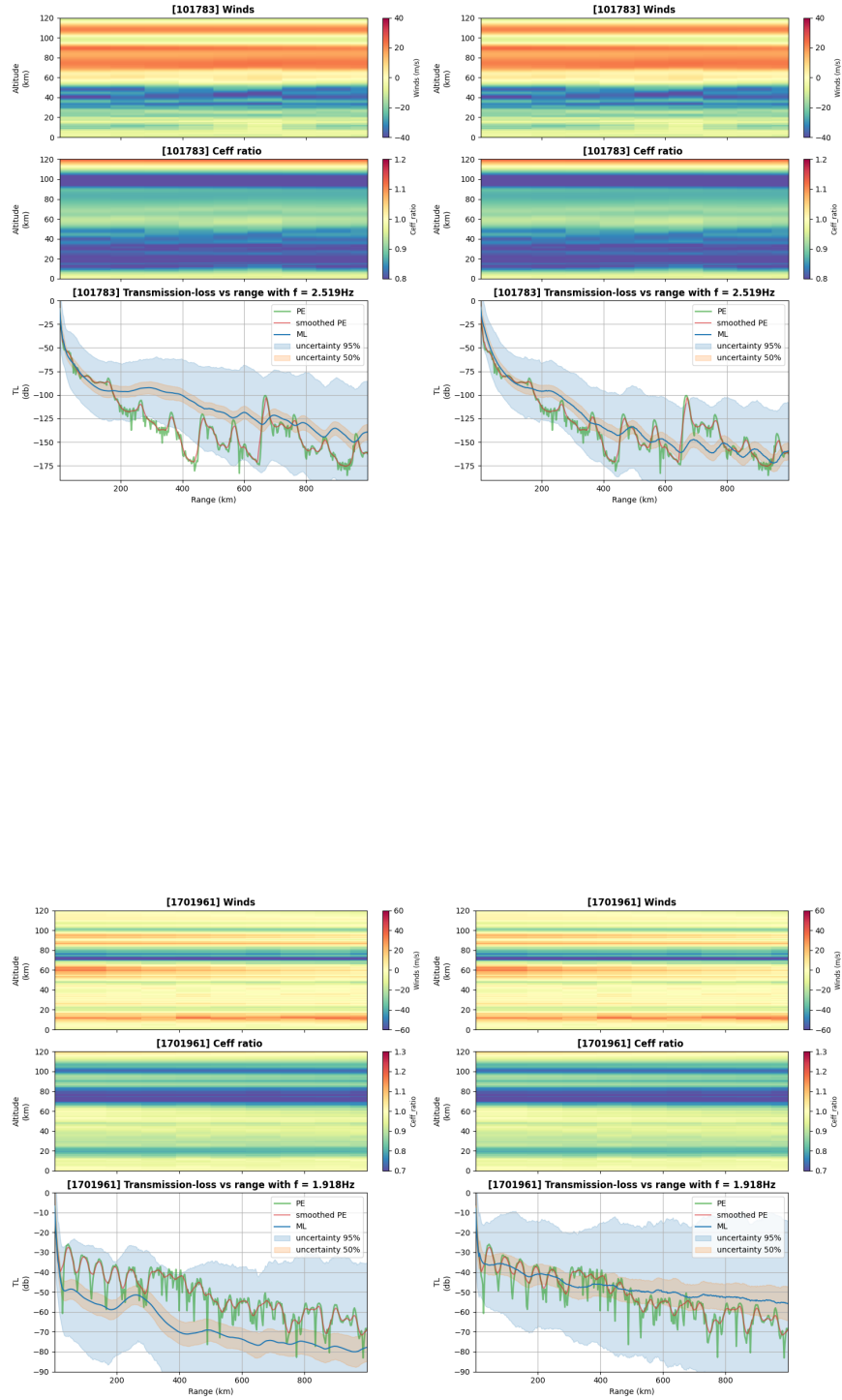


Figure 63: Left: 1D CNN with fully connected, right: 1D CNN with causal CNN

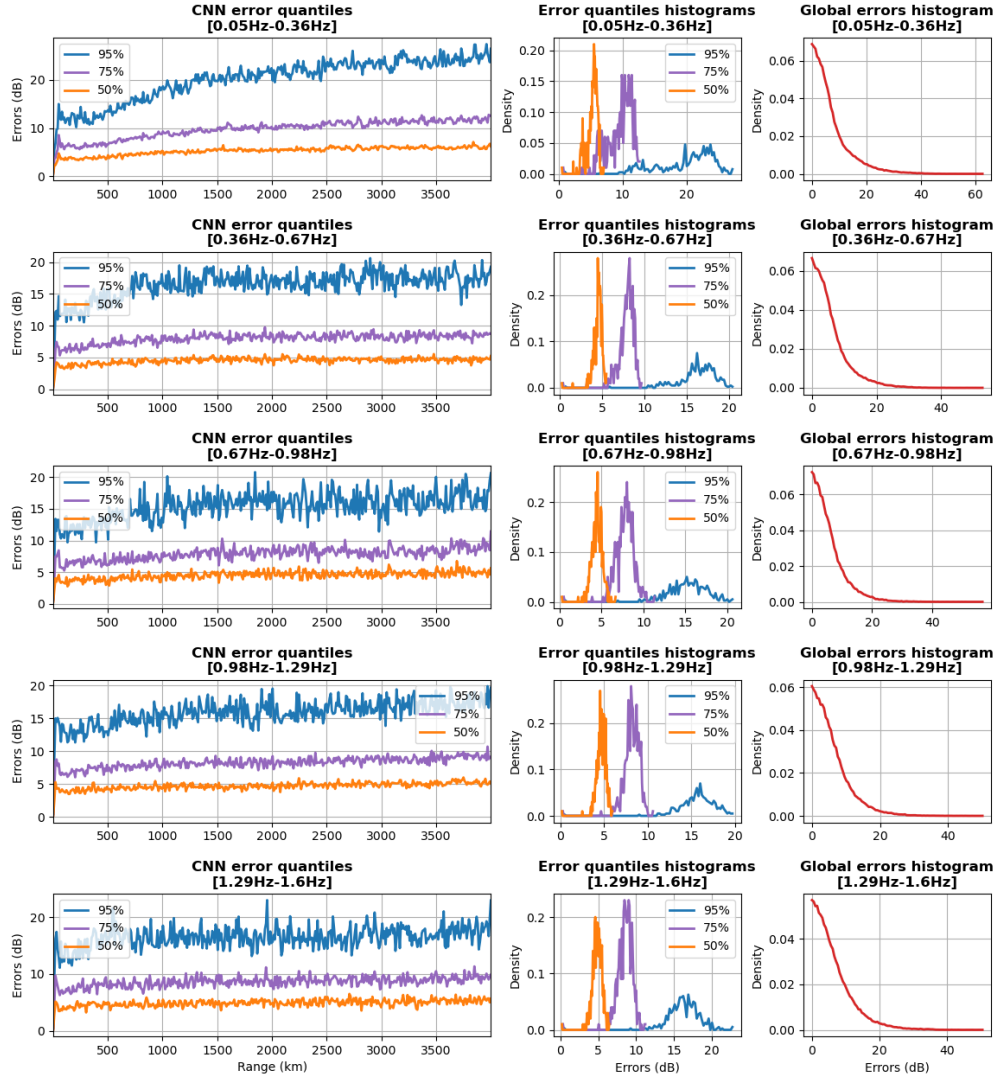


Figure 64: Error summary in various frequency bands (Top to bottom, from 0.05 Hz to 1.6 Hz) for the 2D CNN model over Alexis Le Pichon's 4000 km dataset. Left, RMSE quantiles over the testing dataset vs range (left, in km). Middle, distribution of RMSE quantiles over the testing dataset, and, right, corresponding histogram.

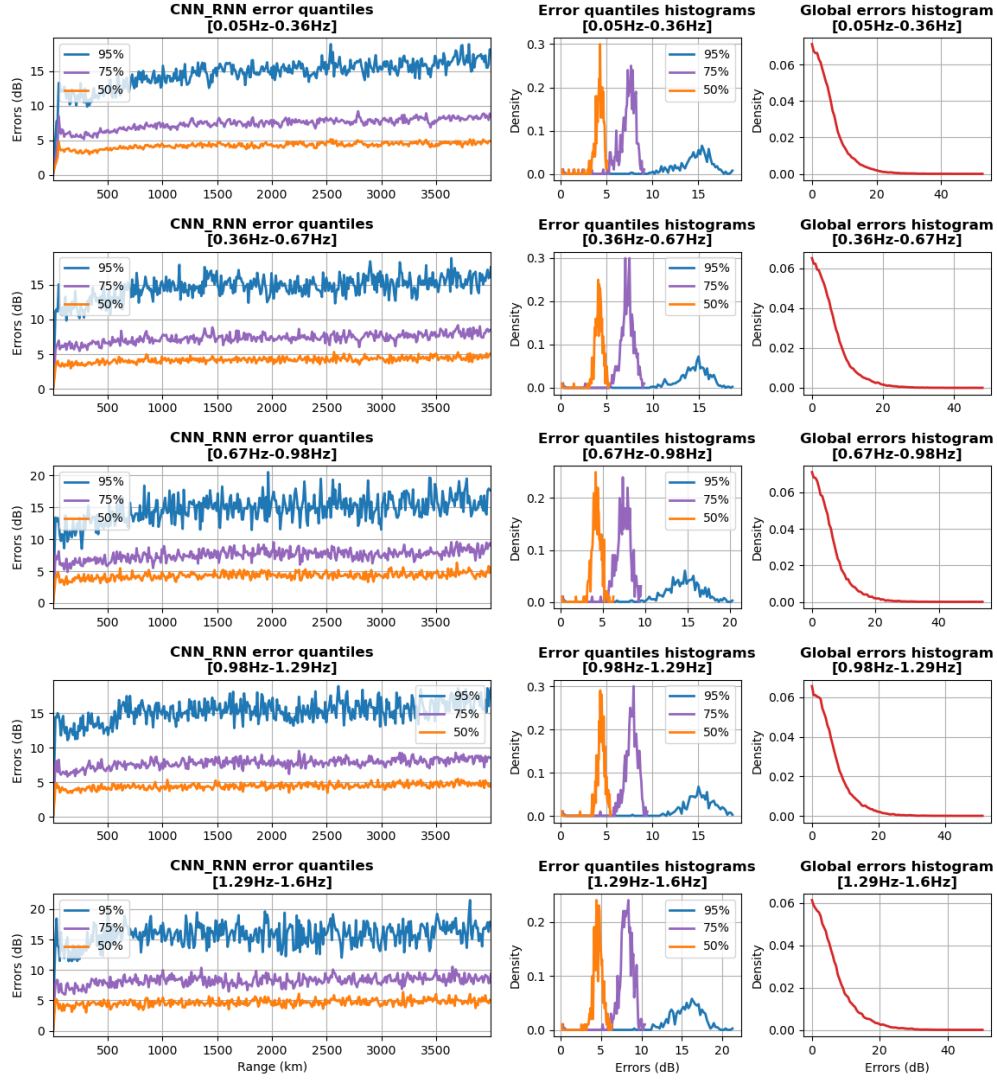


Figure 65: Error summary in various frequency bands (Top to bottom, from 0.05 Hz to 1.6 Hz) for the 1D CNN-RNN model over Alexis Le Pichon's 4000 km dataset. Left, RMSE quantiles over the testing dataset vs range (left, in km). Middle, distribution of RMSE quantiles over the testing dataset, and, right, corresponding histogram.

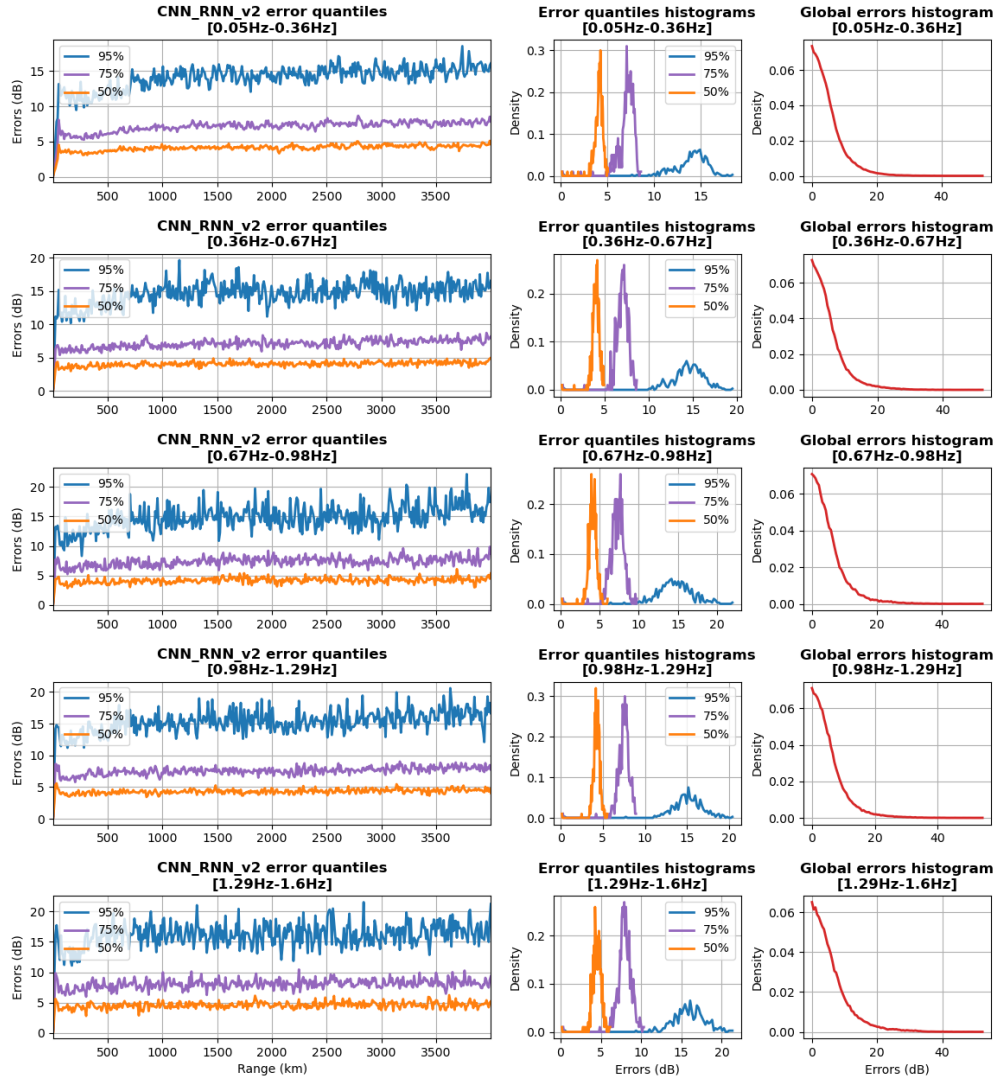


Figure 66: Error summary in various frequency bands (Top to bottom, from 0.05 Hz to 1.6 Hz) for the 1D CNN-RNN with stacked GRUs model (GRU of size 110) model over Alexis Le Pichon's 4000 km dataset. Left, RMSE quantiles over the testing dataset vs range (left, in km). Middle, distribution of RMSE quantiles over the testing dataset, and, right, corresponding histogram.

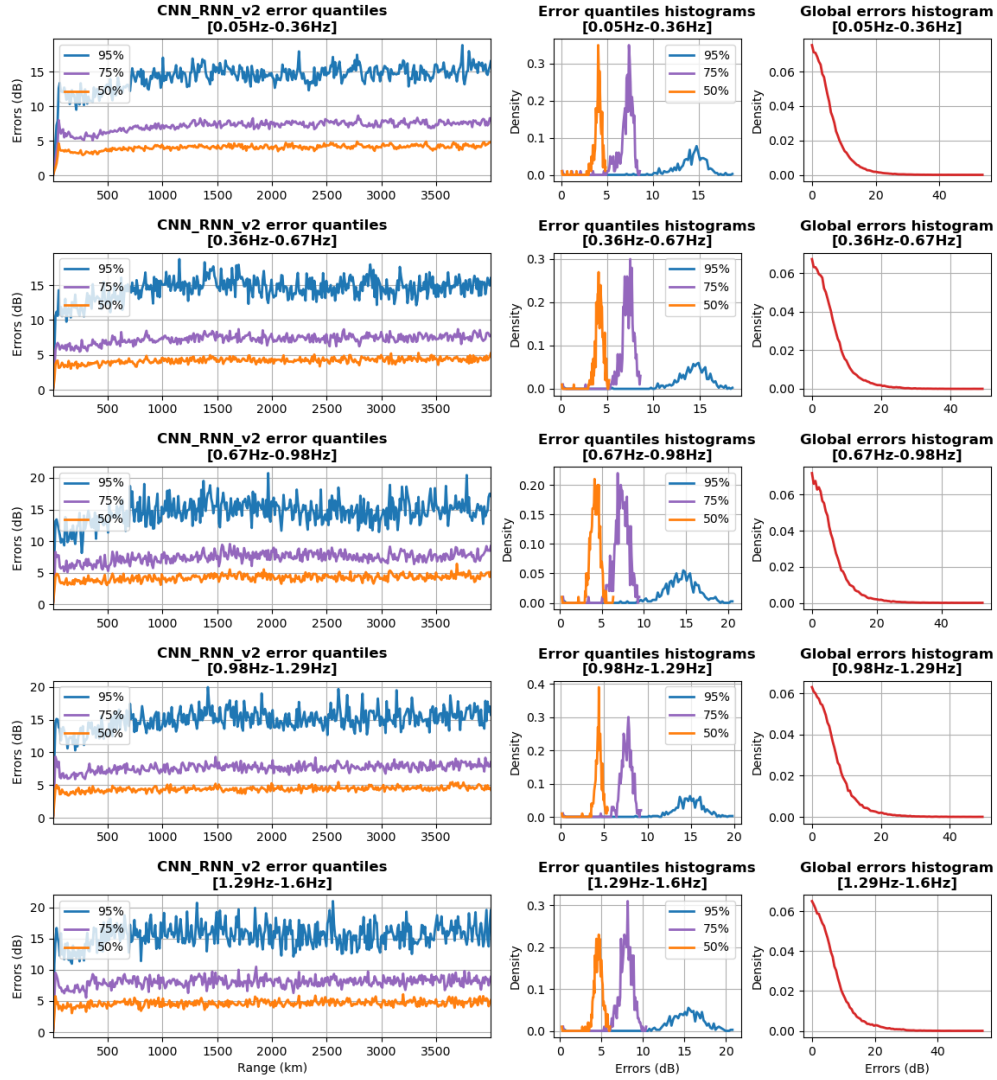


Figure 67: Error summary in various frequency bands (Top to bottom, from 0.05 Hz to 1.6 Hz) for the 1D CNN-RNN with stacked GRUs model (GRU of size 400) model over Alexis Le Pichon's 4000 km dataset. Left, RMSE quantiles over the testing dataset vs range (left, in km). Middle, distribution of RMSE quantiles over the testing dataset, and, right, corresponding histogram.

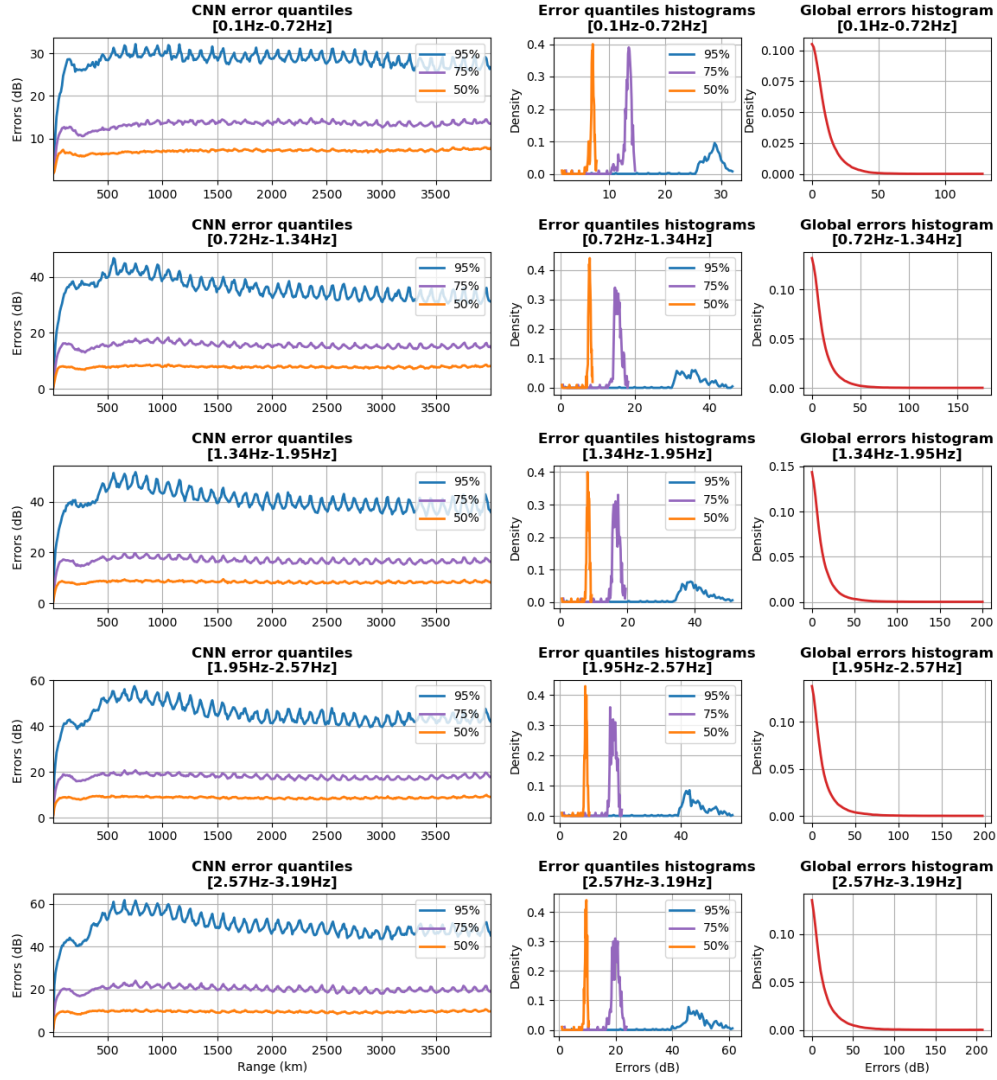


Figure 68: Error summary in various frequency bands (Top to bottom, from 0.05 Hz to 1.6 Hz) for the 2D CNN model over the new 4000 km dataset. Left, RMSE quantiles over the testing dataset vs range (left, in km). Middle, distribution of RMSE quantiles over the testing dataset, and, right, corresponding histogram.

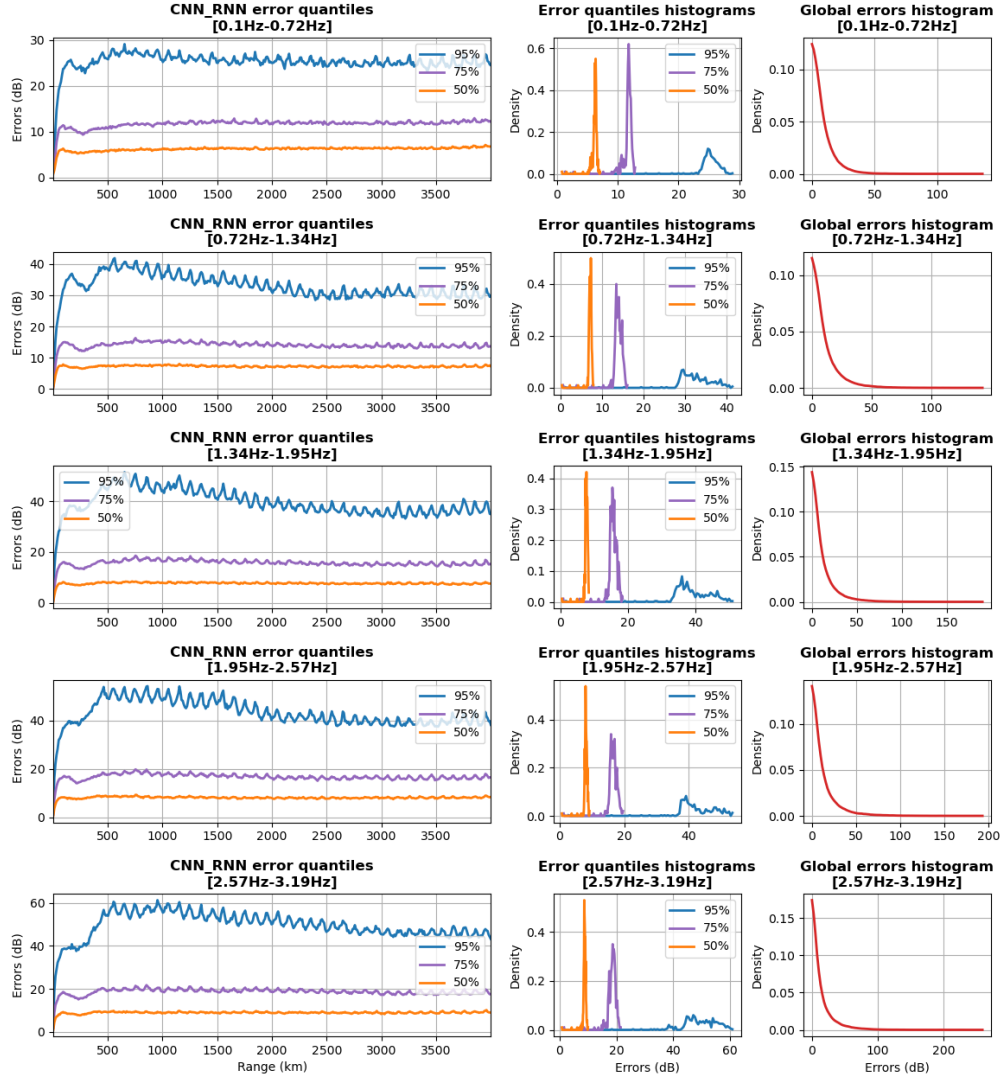


Figure 69: Error summary in various frequency bands (Top to bottom, from 0.05 Hz to 1.6 Hz) for the 1D CNN-RNN model over the new 4000 km dataset. Left, RMSE quantiles over the testing dataset vs range (left, in km). Middle, distribution of RMSE quantiles over the testing dataset, and, right, corresponding histogram.

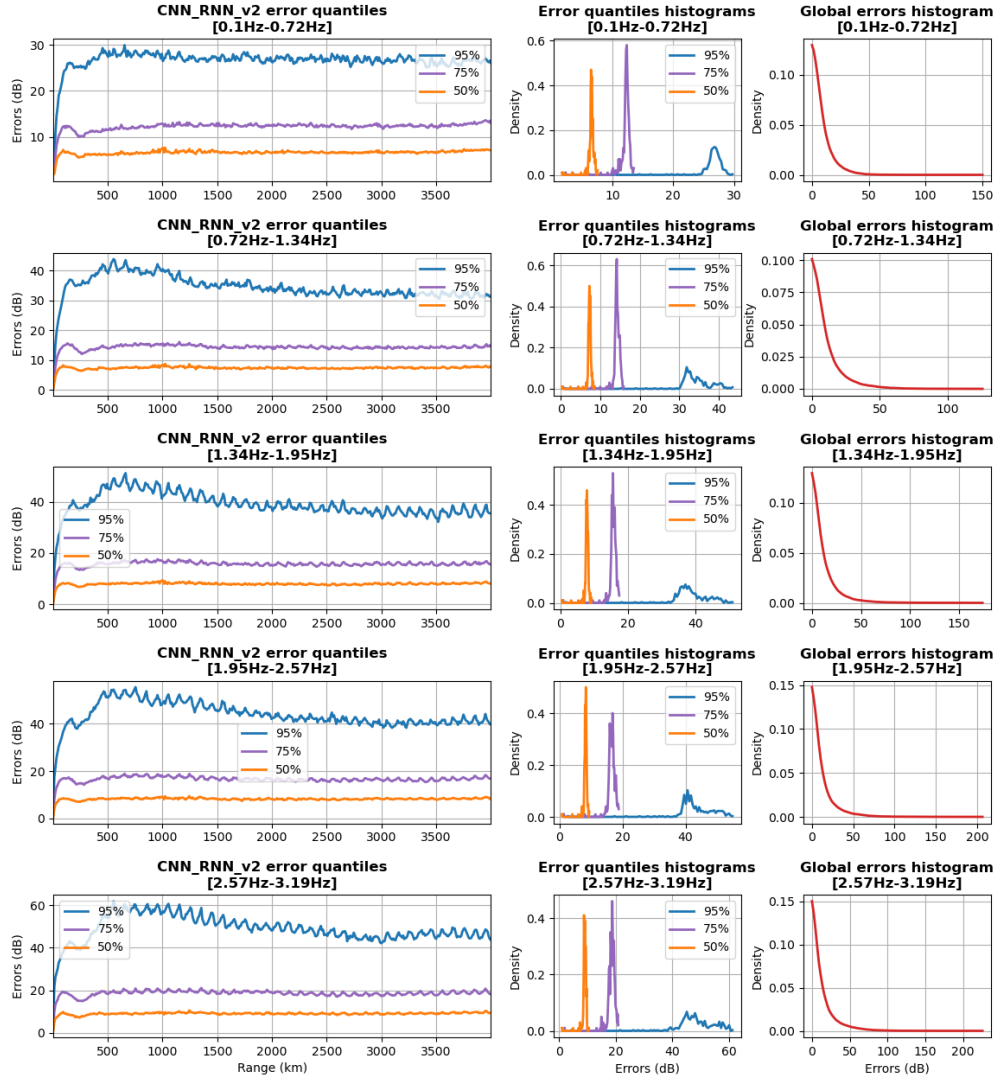


Figure 70: Error summary in various frequency bands (Top to bottom, from 0.05 Hz to 1.6 Hz) for the 1D CNN-RNN with stacked GRUs model (GRU of size 110) model over the new 4000 km dataset. Left, RMSE quantiles over the testing dataset vs range (left, in km). Middle, distribution of RMSE quantiles over the testing dataset, and, right, corresponding histogram.

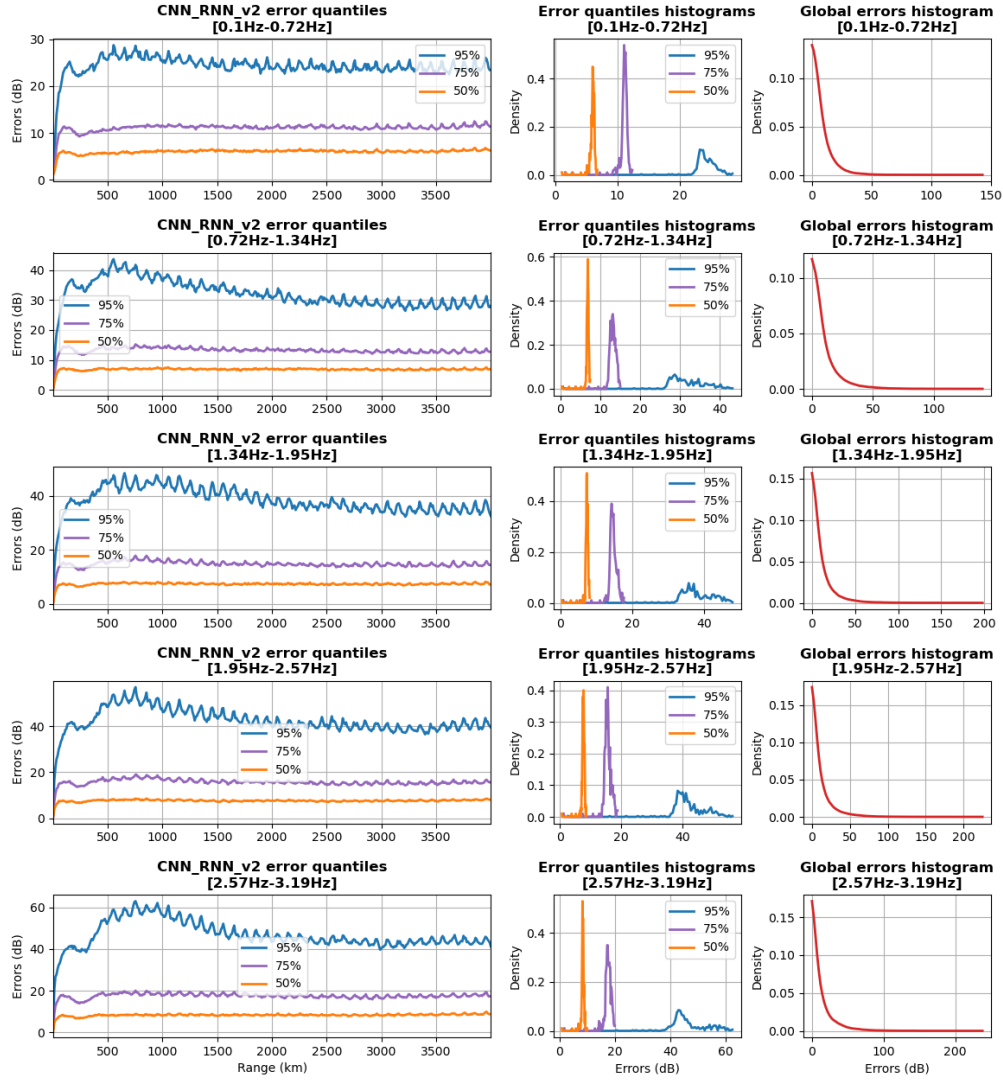


Figure 71: Error summary in various frequency bands (Top to bottom, from 0.05 Hz to 1.6 Hz) for the 1D CNN-RNN with stacked GRUs model (GRU of size 400) model over the new 4000 km dataset. Left, RMSE quantiles over the testing dataset vs range (left, in km). Middle, distribution of RMSE quantiles over the testing dataset, and, right, corresponding histogram.

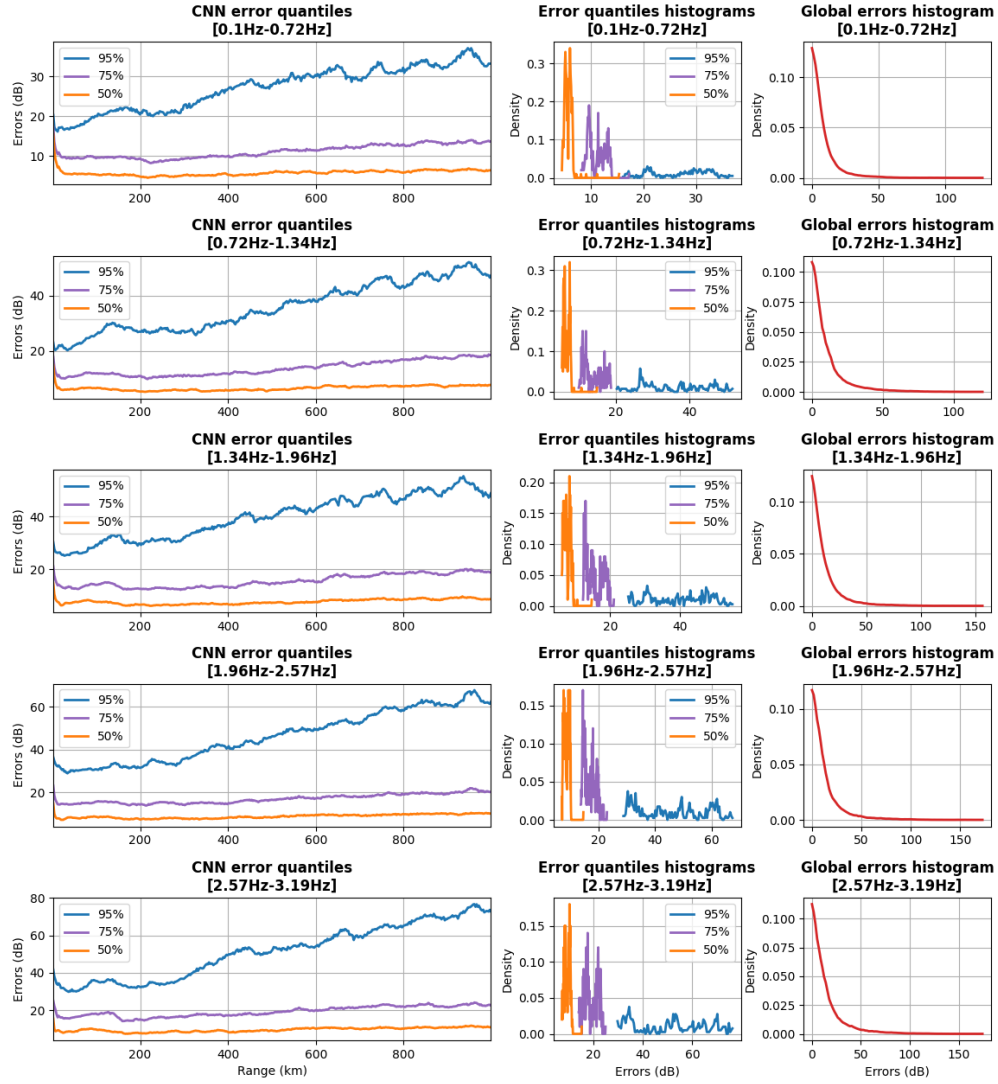


Figure 72: Error summary in various frequency bands (Top to bottom, from 0.05 Hz to 1.6 Hz) for the 2D CNN model over the 1000 km dataset. Left, RMSE quantiles over the testing dataset vs range (left, in km). Middle, distribution of RMSE quantiles over the testing dataset, and, right, corresponding histogram.

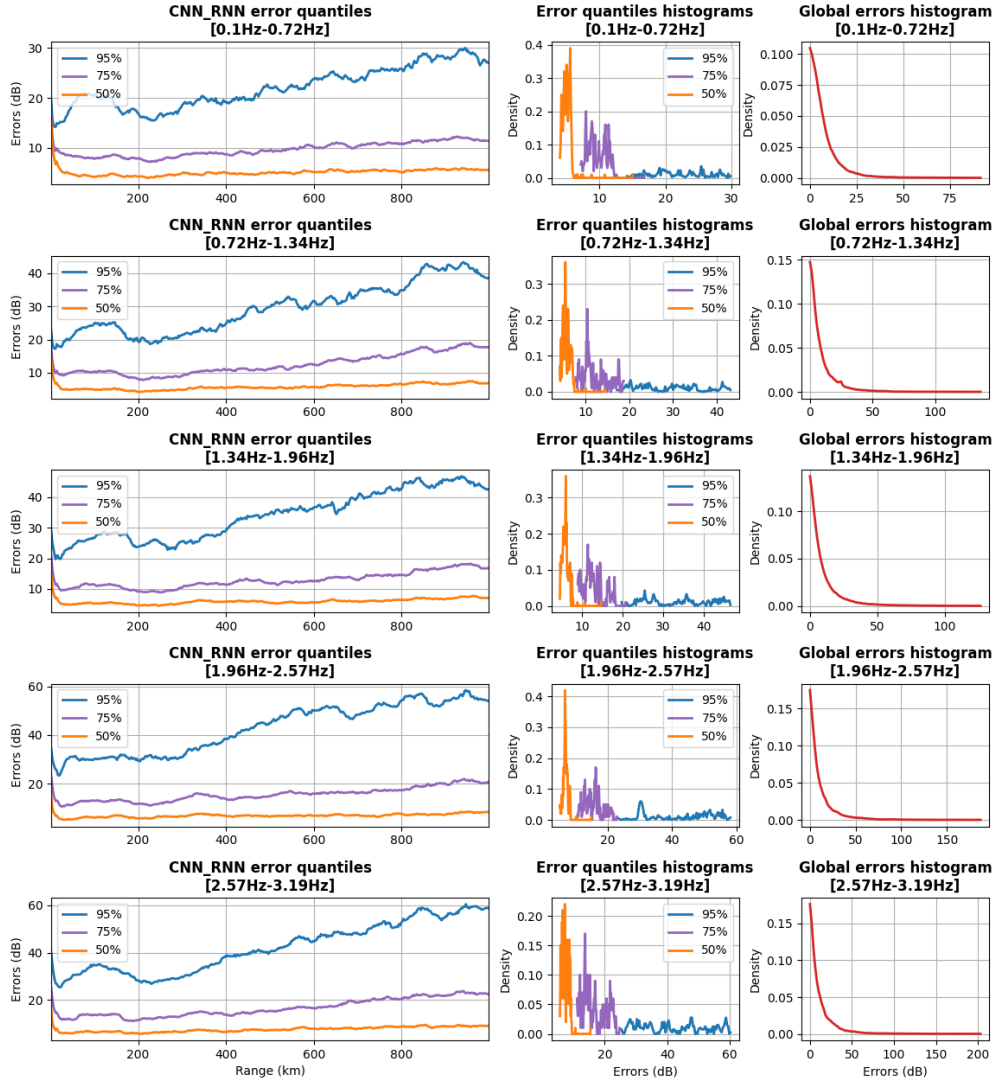


Figure 73: Error summary in various frequency bands (Top to bottom, from 0.05 Hz to 1.6 Hz) for the 1D CNN-RNN model over the 1000 km dataset. Left, RMSE quantiles over the testing dataset vs range (left, in km). Middle, distribution of RMSE quantiles over the testing dataset, and, right, corresponding histogram.

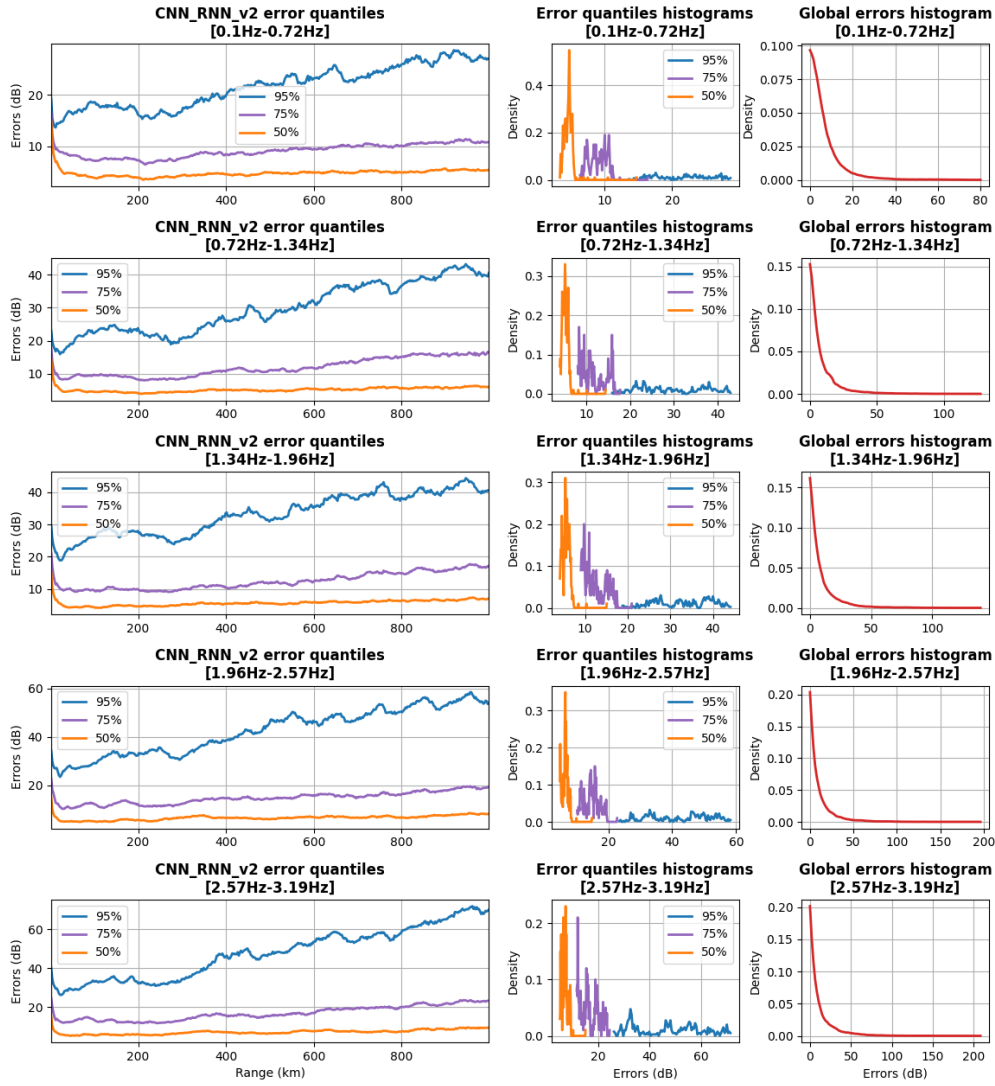


Figure 74: Error summary in various frequency bands (Top to bottom, from 0.05 Hz to 1.6 Hz) for the 1D CNN-RNN with stacked GRUs model (GRU of size 400) model over the 1000 km dataset. Left, RMSE quantiles over the testing dataset vs range (left, in km). Middle, distribution of RMSE quantiles over the testing dataset, and, right, corresponding histogram.

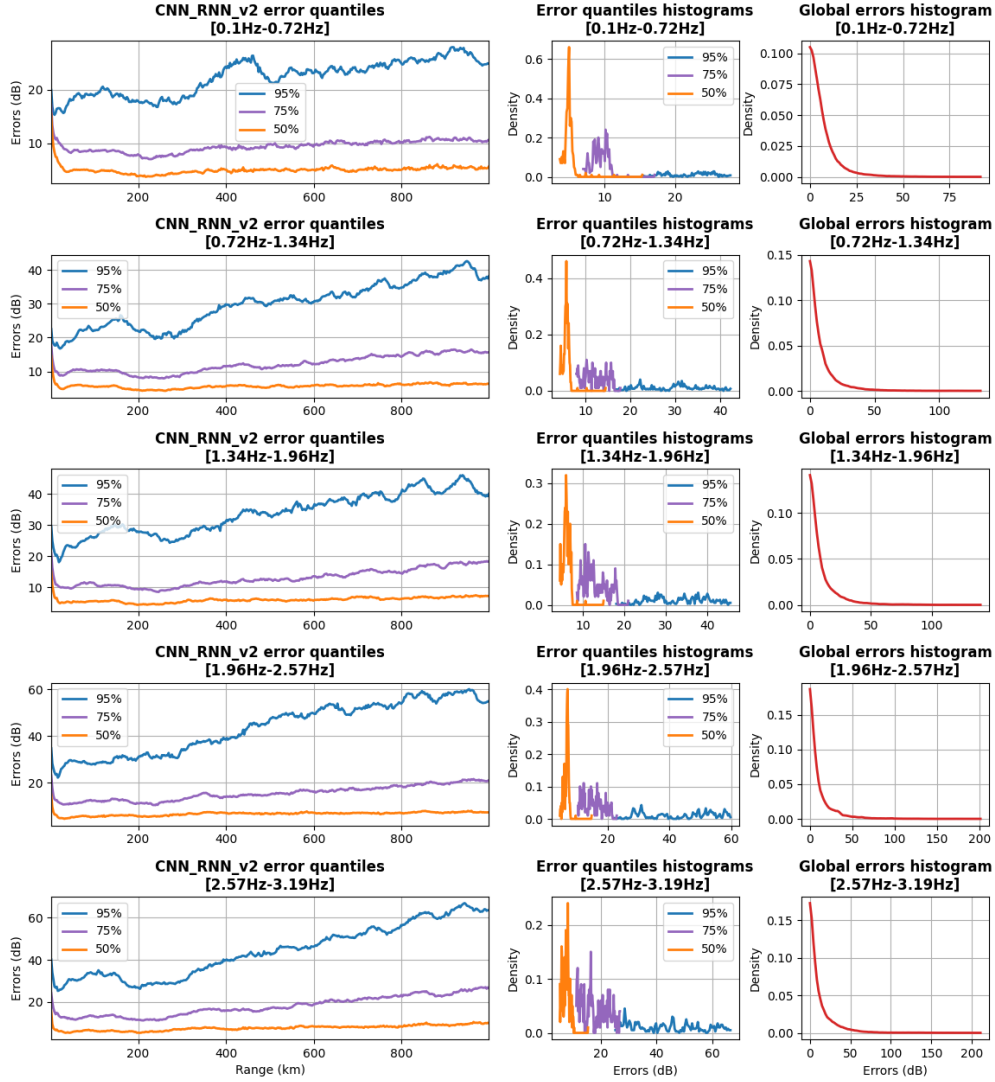


Figure 75: Error summary in various frequency bands (Top to bottom, from 0.05 Hz to 1.6 Hz) for the 1D CNN-RNN with stacked GRUs model (GRU of size 110) model over the 1000 km dataset. Left, RMSE quantiles over the testing dataset vs range (left, in km). Middle, distribution of RMSE quantiles over the testing dataset, and, right, corresponding histogram.