Machine Learning Enabled FBAR Digital Twin for Rapid Optimization

Gergely Simon
OnScale Ltd
Glasgow, United Kingdom
gergely.simon@onscale.com

Gergely B. Hantos
School of Engineering and Physical Sciences
Heriot-Watt University
Edinburgh, United Kindgom
gbh1@hw.ac.uk

Mihir S. Patel
OnScale Ltd
Burlington, MA, USA
patelmihirs@gmail.com

Andrew Tweedie
OnScale Ltd
Glasgow, United Kingdom

Gerald Harvey
OnScale Inc
Redwood City, CA, United States
gerald.harvey@onscale.com

***Abstract*—In this paper we discuss a machine learning-based method to obtain a digital twin of a Thin Film Bulk Acoustic Wave Resonator (TFBAR) that can be used as a surrogate for simulations to estimate resonance frequencies of devices. Normalized root mean square error values better than 0.04% and 0.1% were achieved for 1D and 2D models, respectively. Training times for neural networks were ~20 s for ~2000 epochs and hundreds of datasets.**

Keywords—FBAR, machine learning, regression model, neural network

I. Introduction

TFBAR are commonly used for RF filtering applications in various configurations [1]. Different key performance indicators (KPIs) such as series or parallel resonant frequency, Q value, suppression of lateral modes are important for these topologies [2]. Optimization of a structure is a time-consuming process that is usually performed serially in a feedback loop. Either theoretical, or numerical models are evaluated for a given structure, that are used to inform changes required and feed back to the beginning of the design circle. Simulation speed is typically a bottleneck and addressing this would result in a significant reduction of the time required to generate a design.

We suggest a framework that utilizes efficient parallel execution of hundreds of simulation models, followed by a machine learning (ML) stage for relevant feature extraction and KPI optimization.

II. Device Overview and Methods

A. Device overview

As the thickness of a usual TFBAR structure is much smaller than its lateral extents, useful information of the resonant behavior can be obtained even when using a thin cutout of the geometry with quasi-infinite boundaries, thus forming a 1D model. The 1D model comprised a piezoelectric layer sandwiched between two molybdenum electrodes (Fig. 1).

We decided to remove any passivation from the top of the structure. Furthermore, as the air gap isolates the resonator structure from the substrate, this does not need to be modeled for the 1D case. Symmetry boundaries are employed at the minimum and maximum X-extents. The width of the model is the minimum mesh size for simplicity and the fictious depth of the model was chosen to result in a total static impedance around 50 Ω at resonance. The active area is therefore assumed to be ~0.0144 mm^2.

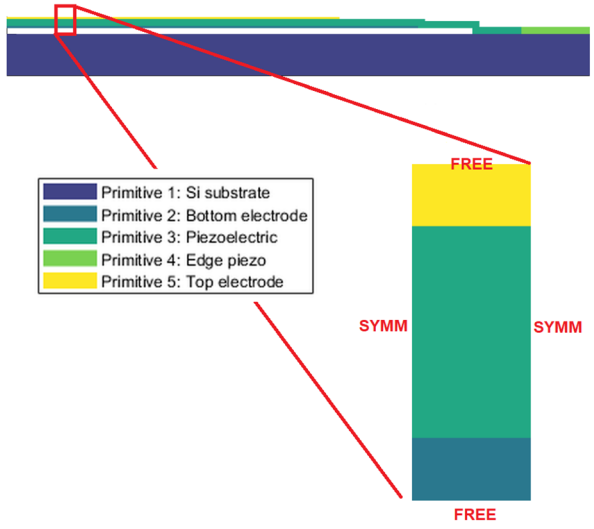

Fig. 1. A 2D TFBAR, and the 1D equivalent model. Note that the transverse dimension of the 2D TFBAR is not to scale.

B. Material properties

For the piezoelectric layer a transverse isotropic deposited aluminum nitride (AlN) was selected (with isotropy plane XY and out-of-plane axis Z), with stiffness matrix values 345 GPa, 395 GPa, 125 GPa, 120 GPa and 118 GPa, for C_{11}, C_{33}, C_{12}, C_{13} and C_{44}, respectively [3]. The remaining values either follow from the commutativity of indices X and Y, and $C_{66} = (C_{11} - C_{12}) / 2$. All the values assume engineering strains. The piezoelectric coupling tensor values were selected as $e_{15} = e_{24} = -0.48$ C/m^2, $e_{31} = e_{32} = -0.58$ C/m^2 and $e_{33} = 1.55$ C/m^2. The dielectric constants are 8 in-plane and 9.5 out-of-plane, and the density 3260 kg/m^3.

978-1-7281-5448-0/20/$31.00 ©2020 IEEE

Molybdenum (Mo) was selected as electrode material with 10022 kg/m^3 density and 284 GPa bulk, 126 GPa shear modulus, resulting in 6650 m/s dilatational and 3509 m/s shear wave speed.

For simplicity, the acoustic quality factor for AlN and Mo is assumed to be Q = 500. This is equivalent to damping constant (η) of 0.002, or damping ratio (ζ) of 0.001.

C. Simulation tool

To be able to execute models in parallel, we used OnScale as the finite element (FE) simulation platform. For 2D models a single run requires ~15 minutes execution time. However, OnScale allows launching these on high performance compute clusters on the cloud, thus the results to all simulations are available after the 15 minutes runtime. Approximately 100 simulations would be only available after a day of local simulations at the same rate. The lightweight 1D models were run locally. In all cases, a time domain method was employed.

D. Machine learning platform

As Python offers freely accessible libraries for machine learning (ML), we decided to use that for training the ML models. A Jupyter notebook front-end was used for rapid development and prototyping, with a Keras – Tensorflow back-end. Regression models which were tested are:

(i) Simple linear regression, (ii) Ridge regression without hyperparameter tuning, (iii) Ridge regression with grid search hyperparameter tuning, (iv) Lasso regression with grid search hyperparameter tuning and (v) Elastic net.

To investigate the applicability of other approaches, random forest and decision tree models were also included in the analysis. Finally, neural networks were anticipated to have the best performance for the problem [4-5].

III. Results

In this section we first present the considerations needed to be made to have accurate results for the ML algorithms and then present the performance of the regression models.

A. Runtime selection

Usually it is recommended that the model is to be run for at least twice as many cycles as the Q value to make sure that minimal residual energy stays in the system and therefore the results would not be significantly affected, had the runtime chosen longer. To make absolutely certain that this is the case, the following investigation was carried out: a model with 35 elements per wavelength (EPW) meshing (resulting in minimum mesh resolution of 63.8 nm) was run for 10,000 cycles, and was considered to be the reference model. Afterwards, the impedance curve was obtained for the 10,000 cycle run. Next, shorter runs were simply obtained by cutting the first N samples of the reference run. Absolute value of impedance curves was interpolated (using spline) onto the reference frequency array, and an RMSE was calculated as the error measure, on the logarithmic data. The logarithmic normalization was included to weight the impedance values both around series (f_s) and parallel (f_p) resonance frequency equally.

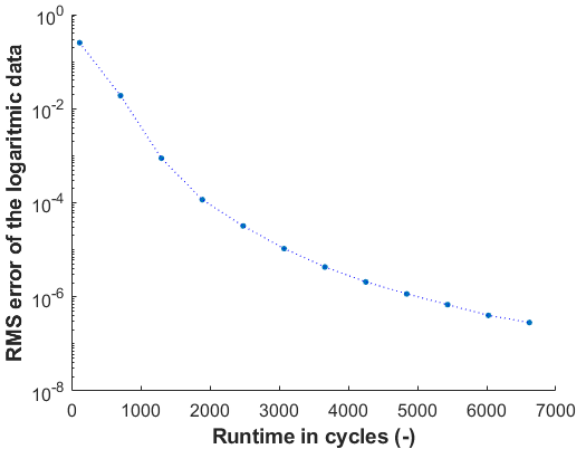

Fig. 2. Investigation of required runtime to maintain precision of results.

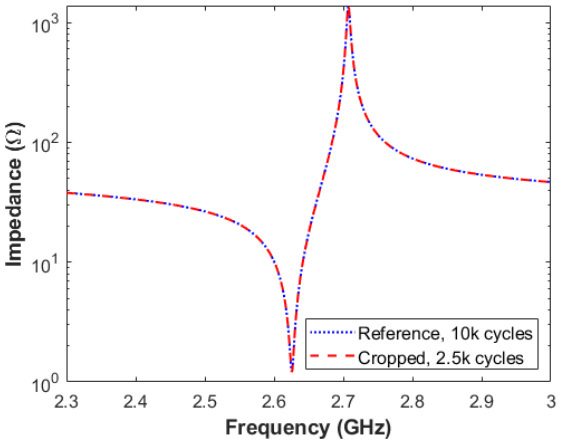

Fig. 3. Impedance of the reference model, using 10,000 cycles for runtime, and the selected 2,500 cycle model. Precision is maintained.

Based on the results of Fig. 2, 2500 cycles runtime was chosen, that results in less than 0.1% error even around f_p (see Fig. 3). All further simulations and investigations were carried out using 2500 cycle runtime. The frequency of interest was taken as 2.688 GHz, so the runtime absolute value is 0.93 us.

B. Mesh convergence study

For the mesh convergence study, the relative error in f_s and f_p was used as error measure. A mesh size of 100 EPW (22.3 nm) was used as the reference model, and all other models were compared to this. To increase precision, the absolute impedance arrays were interpolated onto the reference frequency base as for the runtime selection tests. As the results show (Fig. 4), above 50 EPW mesh size, there is no significant change in results, and for 30-40 EPW mesh size, the relative error is already below 0.1%. Therefore, 35 EPW was chosen as the mesh size, that provides results with relative error around 0.05%.

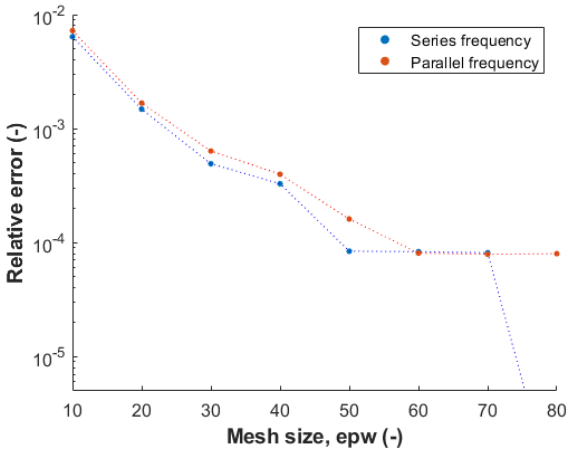

Fig. 4. Mesh convergence tests. The reference model used 100 EPW meshing and relative error of the resonance frequencies is compared to this. 35 EPW offers ~0.05% relative error and was selected as the mesh.

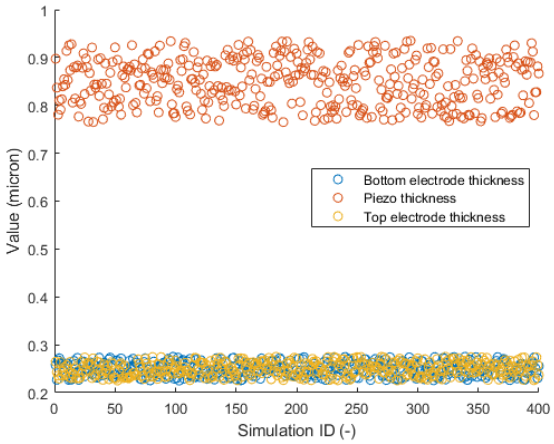

Fig. 5. Example distribution of layer thicknesses for the 400 runs. As the figure illustrates, these all follow linear distribution instead of Gaussian normal distribution.

C. *Generating and processing data*

The nominal 1D device thicknesses were 0.25 µm for the electrodes and 0.85 µm for the piezoelectric layer, resulting in a resonance frequency of about 2.7 GHz. All these parameters were varied according to a linear random distribution between 0.9 to 1.1 times the nominal value, see Fig 5. The selected distribution is favored over a normal distribution, as risk of overfitting and biasing the ML model is reduced. The resulting devices were simulated (15 sec. runtime each) and the f_s and f_p resonance frequencies extracted. For the 1D model a total of 400 simulations were carried out, already resulting in good AI training dataset. For each case, the data was split in a 70%-30% ratio between training data and validation dataset.

D. *Comparison of regression models for 1D TFBAR dataset*

Regression models' performance were evaluated using a normalized root mean square error measure, expressed in a percent value (normalized by the frequency of interest, 2.688 GHz). An overall comparison in shown in Fig. 6. The time required to train the various models is shown in Fig. 7.

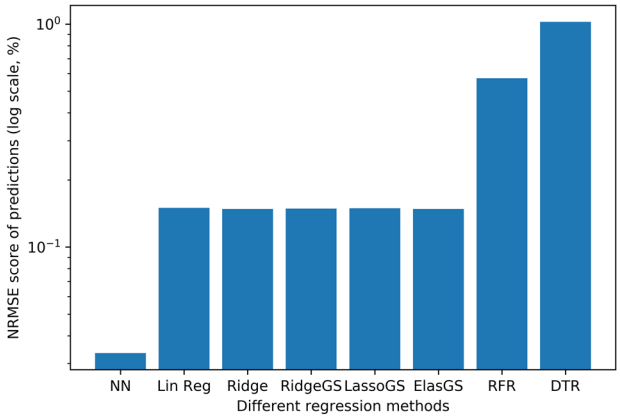

Fig. 6. Normalized root mean square error of various regression methods. The worst performance is of random forest and decision tree, followed by linear regression, and finally neural network with the best performance.

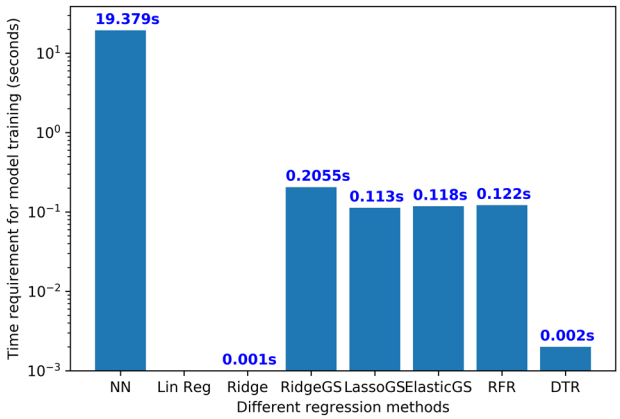

Fig. 7. Comparison of training times for various regression methods. Simple linear regression is <0.001 s, and the neural network is ~20 s. The models with hyperparameter tuning are around 0.1-0.2 s.

Linear regression-based models all have similar, medium performance. They train quick, but exhibit a slightly curved response, as the flat hyperplane cannot predict the real non-linear response accurately (Fig. 8).

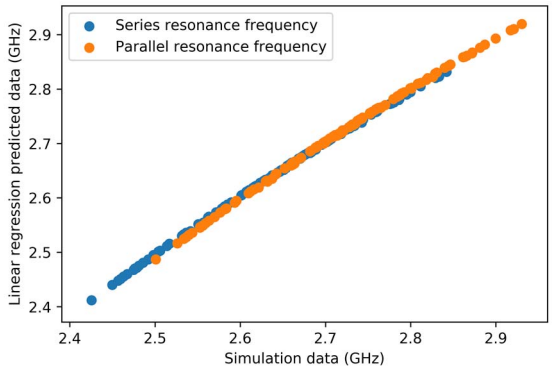

Fig. 8. An example linear regression model result for 1D digital twin. Note the slight curvature of the approximation.

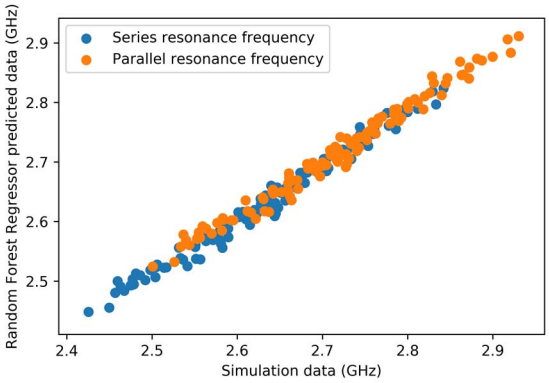

Fig. 9. An example random forest regression model result for 1D digital twin. Note the generally noisy response, as the ML tool cannot pick up the regression task properly.

The two tree-based (decision tree and random forest) approaches cannot accurately pick up the regression task, and therefore have a visibly noisy cross-correlation with the simulation data (Fig. 9), with larger error (although this seems to be around 1%, it can be quite significant for resonator design). The training speed is below a second.

The optimal neural net configuration was found to be a 3-15-10-10-2 topology (3 input neurons, 2 output neurons and 3 hidden layers), with excellent prediction both visually and numerically (below 0.05%). The training times are slightly longer (20 sec for 2000 epochs), but the perfect results compensate for the longer training time (that is the same as executing the model once). During training, the usual 70%-30% training-test data split was used.

E. Neural net performance for 2D models

Finally, the neural network was tested for the 2D TFBAR model (Fig. 1). The 1D models informed us that the neural net outperformed all other methods significantly and possibly a reduction of data is feasible to generate AI regression models with similar accuracy. Therefore, for the 2D dataset, we decided to drop the input data by 75% to 100 only. The models had

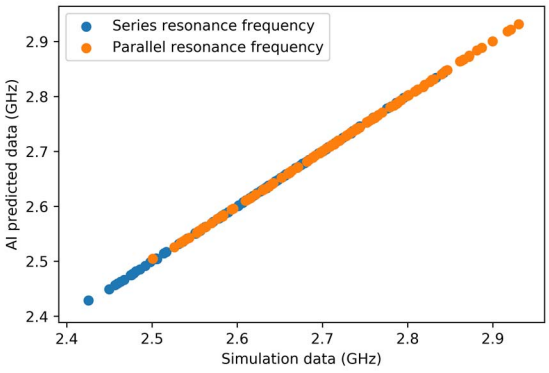

Fig. 10. Results for the neural network regression model for 1D digital twin. Note the visually excellent match between predicted and real data.

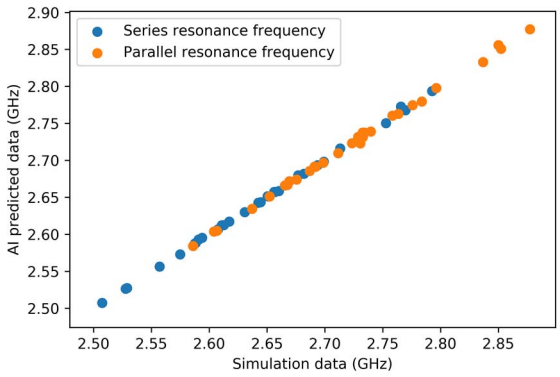

Fig. 11. Results for the neural network regression model for 2D FBAR digital twin. Note the visually excellent match between predicted and real data, with a few outliers that possibly could be improved by an increased dataset.

slightly reduced lateral meshing (to 25 EPW) and ran in 15 minutes on an 8-core machine.

The neural net performance (better than 0.1%) is not significantly better than other approaches: possibly a larger dataset would be ideal.

IV. Conclusions

We investigated generating AI models for accurate prediction of parallel and series resonance frequencies of FBAR devices. To reduce model size, 1D and 2D simulations were carried out. Three input parameters were varied: electrode thicknesses and piezoelectric layer thickness. The results reveal that neural nets outperform all linear regression or tree-based methods, at the cost of a longer training time. Normalized root mean square error values better than 0.04% and 0.1% were achieved for 1D and 2D models, respectively.

The method presented in this paper provides the foundation to extend it to a large suite of geometry and material input parameters to create a general FBAR digital twin that can be used as a surrogate for numerical simulations.

References

[1] M. El Hassan, E. Kerherve, Y. Deval, K. Baraka, J.B. David and D. Belot, “Techniques for Tuning BAW-SMR Resonators for the 4th Generation of Mobile Communications”, 2013, DOI: 10.5772/55131

[2] Y. Liu, Y. Cai, Y. Zhang, A. Tovstopyat , S. Liu and C. Su, “Materials, Design, and Characteristics of Bulk AcousticWave Resonator: A Review”, Micromachines, 2020, 11, 630; doi:10.3390/mi11070630

[3] K. Tsubouchi, K. Sugai, N. Mikoshiba, “AlN Material Constants Evaluation and SAW Properties on AlN/Al2O3and AlN/Si, 1981 Ultrasonics Symposium, DOI: 10.1109/ULTSYM.1981.197646

[4] J. Chen et al., “A comparison of linear regression, regularization, and machine learning algorithms to develop Europe-wide spatial models of fine particles and nitrogen dioxide”, Environment International, Vol 130, 104934, https://doi.org/10.1016/j.envint.2019.104934

[5] J. Schmidt, M. R. G. Marques, S. Botti and M. A. L. Marques, “Recent advances and applications of machine learning in solid-state materials science”, npj Comput Mater 5, 83 (2019). https://doi.org/10.1038/s41524-019-0221-0