Neural Modal Propellant Gauging: A Machine Learning Approach to Microgravity Liquid Propellant Estimation

Henry Martin

Golda Meir High School - Milwaukee - WI

Abstract

Propellant mass gauging remains a critical challenge in spaceflight. Modal Propellant Gauging (MPG) was created to help solve this problem, but in order to use it, a tank must first be calibrated and fingerprinted - a time-consuming and complex process that prohibits new and unknown tank configurations from being used. This research investigates the application of machine learning techniques to improve propellant volume estimation and generalizability by analyzing modal frequencies generated through tank excitation. Five neural network architectures were systematically evaluated using data from the MPG-FOSS project, an existing experimental MPG configuration, with a model emerging as the most promising approach by extracting the top n most powerful modal frequencies of a tank resonated with white noise, referred to in this paper as the "Top-n" model. Achieving 97.5% accuracy and the lowest mean absolute error of 0.025, it demonstrates potential for more versatile and precise propellant measurement across different tank configurations.

Keywords: Propellant Mass Gauging; Modal Analysis; Modal Propellant Gauging

Neural Modal Propellant Gauging: A Machine Learning Approach to Microgravity Liquid Propellant Estimation

Introduction

Accurate propellant gauging has consistently remained a high-priority goal for spaceflight (Stanley, 2005). The challenges of precisely measuring fuel quantities in spacecraft have long been a critical technological hurdle, directly impacting mission planning, safety, and operational efficiency. The current state-of-the-art approach, Modal Propellant Gauging (MPG), has shown significantly higher accuracy than previous methods, but it still has a relatively high (3-4%) margin of error with unsettled sloshing liquids in microgravity and 1-2% error in laboratory tests (Crosby et al., 2019).

MPG's results are achieved by excitation of a tank with white noise and then measurement of the modal frequencies that result from this excitation (Crosby et al., 2015, 2019). The fundamental principle relies on understanding how the tank's natural resonant frequencies change as the propellant volume and distribution within the tank are modified. These modal frequencies serve as a acoustic fingerprint of the tank's internal liquid state, offering a non-invasive approach to propellant measurement. The above widely accepted Spectral Density approach to determining a tank's fill level from these modal frequencies suffers from significant limitations that prevent general applicability to previously unknown tanks, requiring specialized, time-intensive fingerprinting to correctly work on new tanks. Specifically, across different tanks, modal frequencies do not shift at the same rate. This inherent limitation reduces the method's versatility and increases the complexity of implementing modal propellant gauging across diverse spacecraft and propulsion systems.

This paper proposes applying machine learning to this problem of new and unknown tanks in place of typical algorithmic approaches and explores multiple neural network architectures, comparing their performance and accuracy on a laboratory benchmark of data.

Data

This research makes use of data collected by the MPG-FOSS (Modal Propellant Gauging via Fiber Optic Sensing System) project at Carthage College, which I have participated in for the last 2 years under the direction of Dr. Kevin Crosby as part of the Summer Undergraduate Research Experience (SURE) program.

MPG-FOSS is configured as shown in Fig. 1. A tank is outfitted with a lead zirconate titanate (PZT) patch, which produces a vibration when electrically stimulated (Viswanath Allamraju & Korla, 2017) and a Fiber Optic Sensing System (FOSS), a system for accurate measurement of strain developed by NASA (Chan et al., 2015). A FOSS works by sending a wide-spectrum laser down a fiber optic with Fiber Bragg Gratings (FBGs) placed intermittently throughout, then measuring the reflected wavelengths to determine microscopic fluctuations in the strain of the FBGs (Chan et al., 2015; Gao et al., 2022). This technology, although extremely useful for its intended purpose of measuring stress and strain, also lends itself to measuring small vibrations, and offers a much less EMI (electromagnetic interference)-prone method of measuring tank vibrations.

The resulting vibration data is recorded alongside weight data from a scale connected to a secondary reservoir outside of the tank as a "ground truth" value for validating fuel gauging accuracy. The raw data from the FOSS is split into 1-second windows and each window has an approximate Power Spectral Density (PSD) calculated for it using *Welch's method*. A raw waveform for a 1-second window is shown in Fig. 2. First, the data is preprocessed and filtered, which is defined as:

Let $X = \{x_1, x_2, ..., x_n\}$ be the input data set.

Let $\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$ be the mean

Let $\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2}$ be the standard deviation

Filtering invalid or dropped frames of data:

$X_{filtered} = \{x \in X : |x - \mu| \leq \sigma\}$

Then, for each 1-second window of 11098 values (x, the sample rate in Hz of the FBG interrogator used) in $X_{filtered}$, calculate the PSD with *Welch's method* (Smith, accessed 2025; Welch, 1967):

Let $s[n]$ be a discrete time series of length N

Split $s[n]$ into K overlapping segments of length L:

$s_k[n] = s[n + kD]$, for $n = 0, ..., L - 1$ where D is the overlap offset

Apply window function $w[n]$ to each segment:

$\hat{s}_k[n] = s_k[n]w[n]$

Calculate periodogram for each windowed segment:

$P_k(f) = \frac{1}{L}|\sum_{n=0}^{L-1} \hat{s}_k[n]e^{-j2\pi fn}|^2$

PSD estimate is the average of $/K$ periodograms:

$P_{welch}(f) = \frac{1}{K}\sum_{k=0}^{K-1} P_k(f)$

Results of these PSD calculations can be seen as a single 1-second window in Fig. 3 and as a surf (3D) graph in Fig. 4. Note the peaks in power at approximately 1800Hz and the increase in frequency over time as the tank drains and the modes shift back to the tank's natural empty modal frequency at ~1100Hz.

Model Architectures

The five models in this paper are all standard multilayer feedforward perceptrons (da Silva et al., 2017) in which a set of inputs is transformed and reduced through a series of hidden layers and finally outputs to a single neuron, which is read as output. These models are optimized through the Stochastic Gradient Descent algorithm (Ruder, 2017) and the Adam optimizer (Kingma & Ba, 2017). These optimizers were chosen for their widespread usage in preexisting work, and the overall architecture of a multilayer perceptron was chosen because of its ability to model somewhat complex relations in noisy data (Stempfel, 2007). Such models can also operate with a relatively small amount of processing power, both in training and runtime inference, allowing them to be easily embedded in processors not unlike what would be sent into microgravity (Ortigosa et al., 2003).

Full-PSD (No Further Processing)

This model takes slices of the entire PSD across all frequencies without any dimensionality reduction or compression, and with no data pre-processing steps besides those described in *Data*. This model has a total of $2ns$ input neurons, where n is the number of slices of the PSD from which samples are taken, and s is the length of each PSD, usually 4700 frequencies. Due to the high number of inputs, this model has a large number of trainable parameters, leading to worse performance and increased power consumption during both training and inference.

Top-n

This model has a total of $4n$ input neurons. Here, n represents the number of frequencies with the highest power from the PSD (see *Data*) that are chosen for sampling, and for the purposes of this paper and all results therein, n is set to 10. These frequencies are then fed into the neural network as input features. Specifically, the model incorporates two sets of input values from the initial phase of sampling, which correspond to a known state or known fill percentage, and two from the present state, which relate to an unknown state or fill percentage. This slicing technique is shown in Fig. 5, and this model architecture is illustrated in Fig. 6.

Auto-encoder

This model is similar to the above models in its architecture, except it makes use of a separate auto-encoder model (also a multilayer feedforward perceptron) to reduce the dimensionality of the PSD before training (Bank et al., 2023). This process makes the number of trainable parameters in the model significantly less than that of the full-PSD approach, while still providing much more depth than the Top-n architecture. This model omits the traditional decoding phase of the auto-encoder and instead trains only two models - one which trains to produce the input to the primary model, and the primary model itself, which transforms the input and produces a single output. The encoder model is depicted in Fig. 7 and the primary model is depicted in Fig. 8.

Top-n Auto-encoder

This model combines the principle of the Top-n model with that of the Auto-encoder model. It is functionally identical to the previous model except it takes only the top n frequencies of the PSD as input to the encoder.

Multi-feature

This model combines 4 forms of encoding into one large model, yielding the following as a large concatenated sequence of input neurons to a much more elaborate model:

- Full raw PSD (Welch's method)
- Encoded PSD as 10 inputs from an auto-encoder (see Fig. 7)
- Top n frequencies in the PSD ($n = 10$)
- Spectral centroid, bandwidth, and flatness (Peeters, 2004)

However, with this much input data, the model is much larger and therefore more computationally expensive to run.

Results & Conclusion

Each model architecture was assessed based on several metrics: accuracy, calculated as the inverse of the mean gauging resolution (Crosby et al., 2019), mean absolute error, R^2 score, training time, and inference time. The results of model performance and inference time are shown in Table 1.

The Top-n model demonstrated the highest accuracy and lowest mean absolute error, making it the most effective at predicting the desired outcome. In addition, it has the shortest inference time, making it suitable for real-time applications. Its training time is relatively short compared to other models due to its simplicity, and can therefore also be easily retrained with further data to increase its accuracy.

The Auto-encoder model, while slightly less accurate than the Top-n model, still maintains a high level of performance. It offers a balance between accuracy and computational efficiency, with a moderate training and inference time, relative to the other models.

Conversely, the Multi-feature model, despite incorporating a comprehensive feature set, showed the lowest accuracy, highest mean absolute error, and longest inference time. This suggests that the complexity of this model may not be justified by its performance, especially considering the computational resources it requires.

Both the Full-PSD and Top-n auto-encoder models exhibited unremarkable performance with significantly worse accuracy relative to those of the other models.

In the future, testing these models against data collected in microgravity experiments would be ideal, as well as training and verifying on more tanks beyond the limited number available for MPG-FOSS. Larger amounts of training data would also be beneficial to the overall performance and accuracy of the models.

Given these results, the Top-n model demonstrates the most promising architecture for modal propellant gauging, providing an optimal trade-off between accuracy and computational cost. Overall, the Top-n model shows an average gauging resolution of 2.5%, which is better than the existing state-of-the-art method of MPG data processing and seems to be a promising potential option for future applications.

References

Bank, D., Koenigstein, N., & Giryes, R. (2023). Autoencoders. In L. Rokach, O. Maimon, & E. Shmueli (Eds.), *Machine learning for data science handbook: Data mining and knowledge discovery handbook* (pp. 353–374). Springer International Publishing. https://doi.org/10.1007/978-3-031-24628-9_16

Chan, H. M., Parker, A. R., Piazza, A., & Richards, W. L. (2015). Fiber-optic sensing system: Overview, development and deployment in flight at nasa. [AFRC-E-DAA-TN25056. Work of the US Gov. Public Use Permitted.]. *Proceedings of the IEEE Avionics and Vehicle Fiber Optics and Photonics Conference (AVFOP 2015)*. https://ntrs.nasa.gov/citations/20150022384

Crosby, K. M., Rundle, T., LeCaptain, K., & Werlink, R. (2015). Modal propellant gauging in low gravity. In *Aiaa space 2016*. American Institute of Aeronautics; Astronautics, Inc. https://doi.org/10.2514/6.2016-5533

Crosby, K. M., Williams, N. J., Werlink, R. J., & Hurlbert, E. A. (2019). Modal propellant gauging: High-resolution and non-invasive gauging of both settled and unsettled liquids in reduced gravity. *Acta Astronautica*, *159*, 499–507. https://doi.org/10.1016/j.actaastro.2019.01.050

da Silva, I. N., Hernane Spatti, D., Andrade Flauzino, R., Liboni, L. H. B., & dos Reis Alves, S. F. (2017). Multilayer perceptron networks. In *Artificial neural networks : A practical course* (pp. 55–115). Springer International Publishing. https://doi.org/10.1007/978-3-319-43162-8_5

Gao, W., Liu, J., Guo, H., Jiang, X., Sun, S., & Yu, H. (2022). Multi-wavelength ultra-weak fiber bragg grating arrays for long-distance quasi-distributed sensing. *Photonic Sensors*, *12*(2), 185–195. https://doi.org/10.1007/s13320-021-0635-4

Kingma, D. P., & Ba, J. (2017). Adam: A method for stochastic optimization. https://arxiv.org/abs/1412.6980

Ortigosa, E. M., Cañas, A., Ros, E., & Carrillo, R. R. (2003). Fpga implementation of a perceptron-like neural network for embedded applications. *Artificial Neural Nets Problem Solving Methods: 7th International Work-Conference on Artificial and Natural Neural Networks, IWANN2003 Maó, Menorca, Spain, June 3–6, 2003 Proceedings, Part II 7*, 1–8.

Peeters, G. (2004). A large set of audio features for sound description (similarity and classification) in the cuidado project.

Ruder, S. (2017). An overview of gradient descent optimization algorithms. https://arxiv.org/abs/1609.04747

Smith, J. O. (accessed 2025). *Spectral audio signal processing* [online book, 2011 edition]. `https://ccrma.stanford.edu/~jos/sasp/`.

Stanley, D. (2005). *Nasa's exploration systems architecture study* (Technical report). National Aeronautics and Space Administration.

Stempfel, G. (2007). Learning kernel perceptrons on noisy data using random projections, 328–342. https://doi.org/10.1007/978-3-540-75225-7_27

Viswanath Allamraju, K., & Korla, S. (2017). Studies on mechanical and electrical characteristics of pzt-5h patch [5th International Conference of Materials Processing and Characterization (ICMPC 2016)]. *Materials Today: Proceedings*, *4*(2, Part A), 126–132. https://doi.org/https://doi.org/10.1016/j.matpr.2017.01.005

Welch, P. (1967). The use of fast fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms. *IEEE Transactions on Audio and Electroacoustics*, *15*(2), 70–73. https://doi.org/10.1109/TAU.1967.1161901

Model Architecture	Accuracy	Mean Absolute Error	R^2 Score	Training Time (sec)	Inference Time (ms)
Top-n	97.5%	0.025	0.985	120	2.5
Auto-encoder	96.8%	0.032	0.973	180	3.7
Top-n Auto-encoder	96.5%	0.035	0.968	210	4.2
Full-PSD	80.2%	0.048	0.952	350	6.5
Multi-feature	73.5%	0.053	0.941	450	8.3

Table 1

Performance metrics of different model architectures.

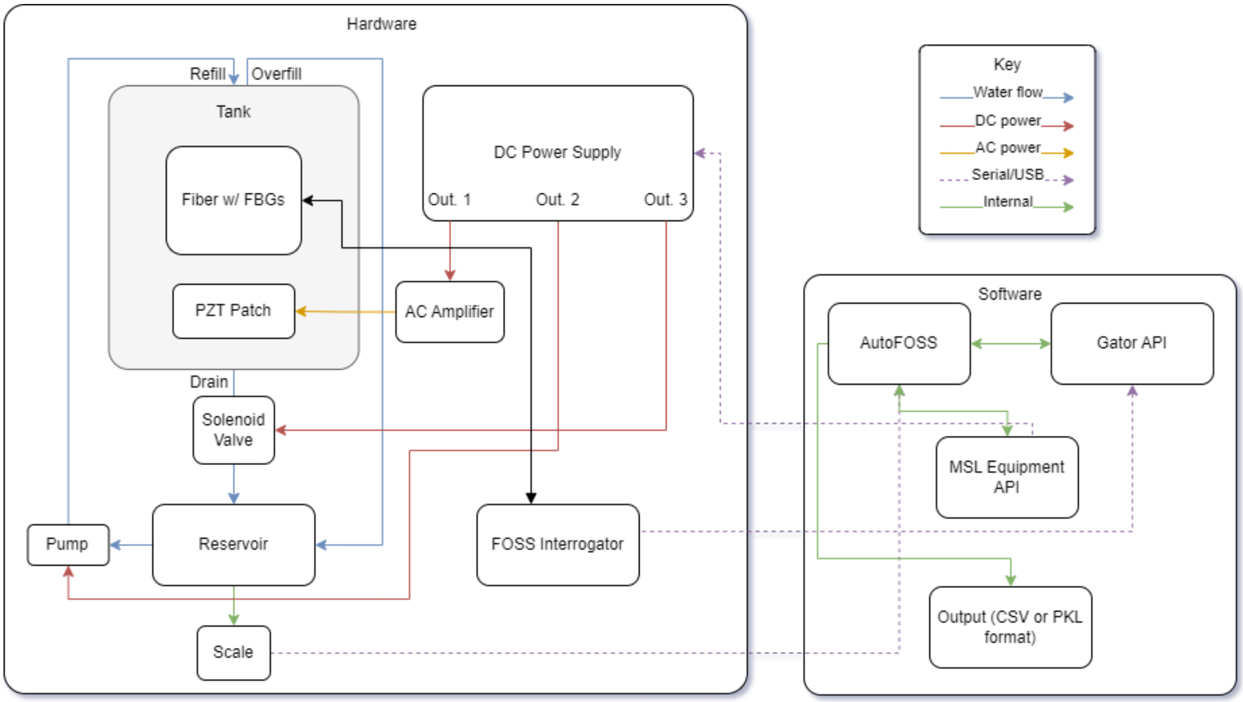

Figure 1

MPG-FOSS Operational Block Diagram

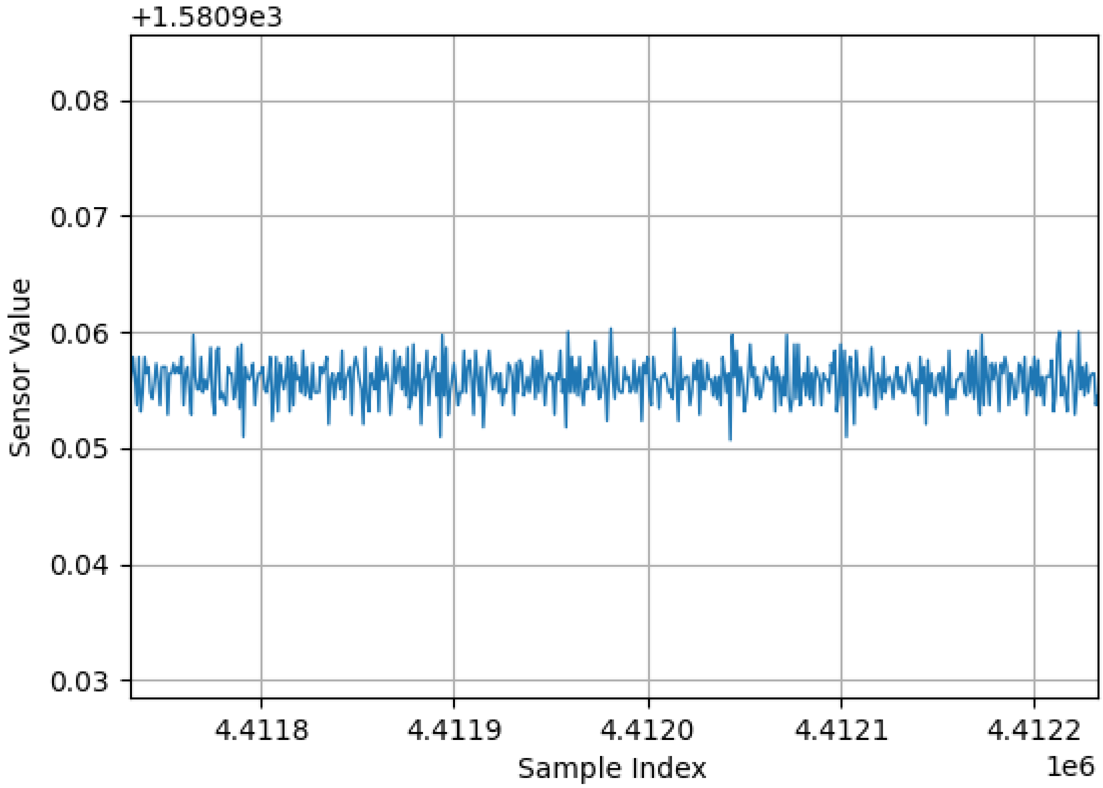

Figure 2

Raw MPG waveform for a 1-second window

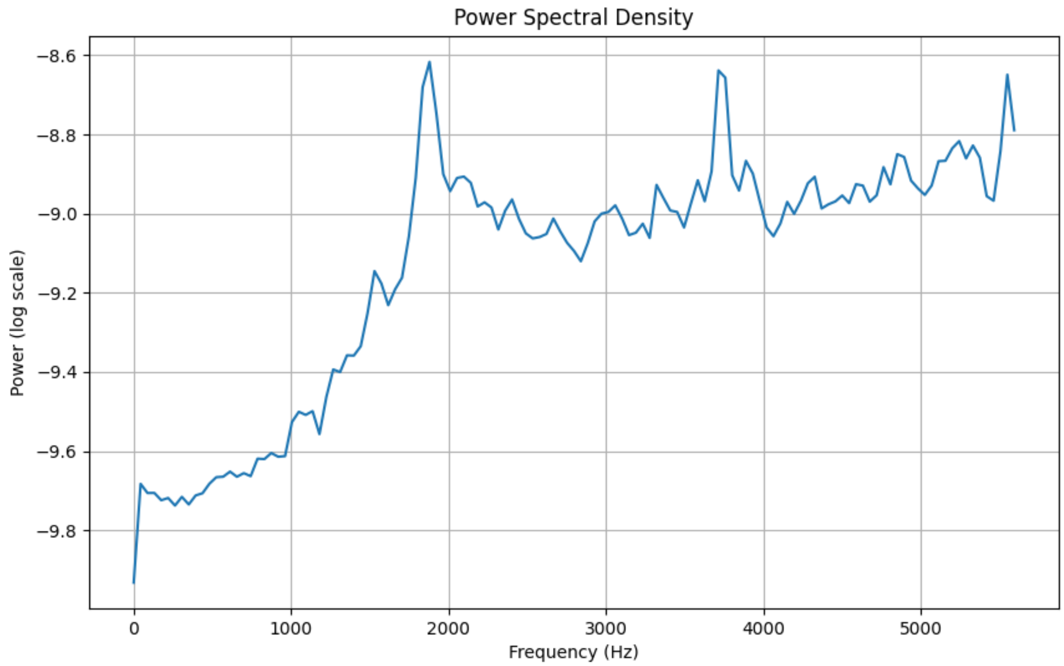

Figure 3

PSD of a 1-second window of MPG data

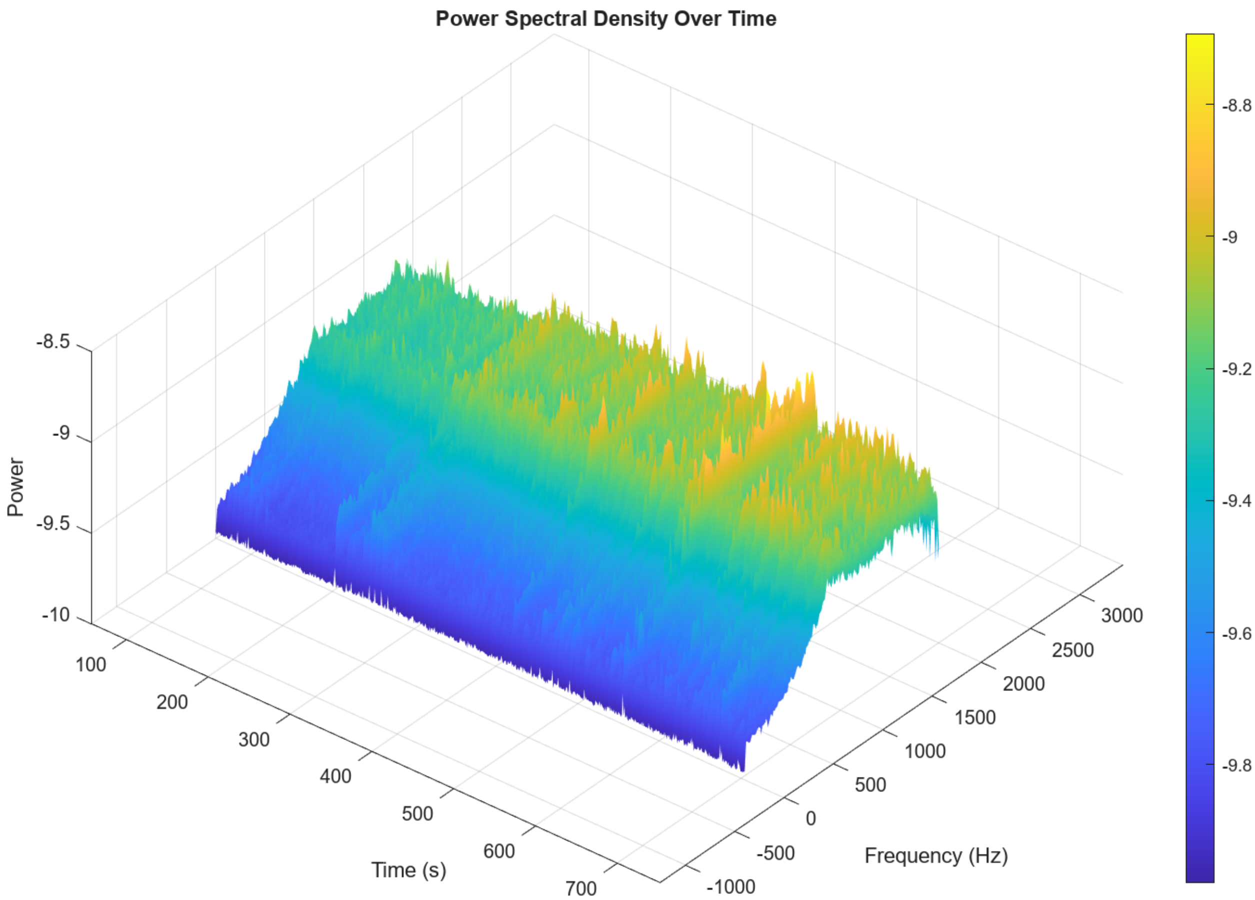

Figure 4

Surf plot of a 10-minute drain on an experimental MPG loop - Note the modal frequencies move upwards in frequency as the time increases

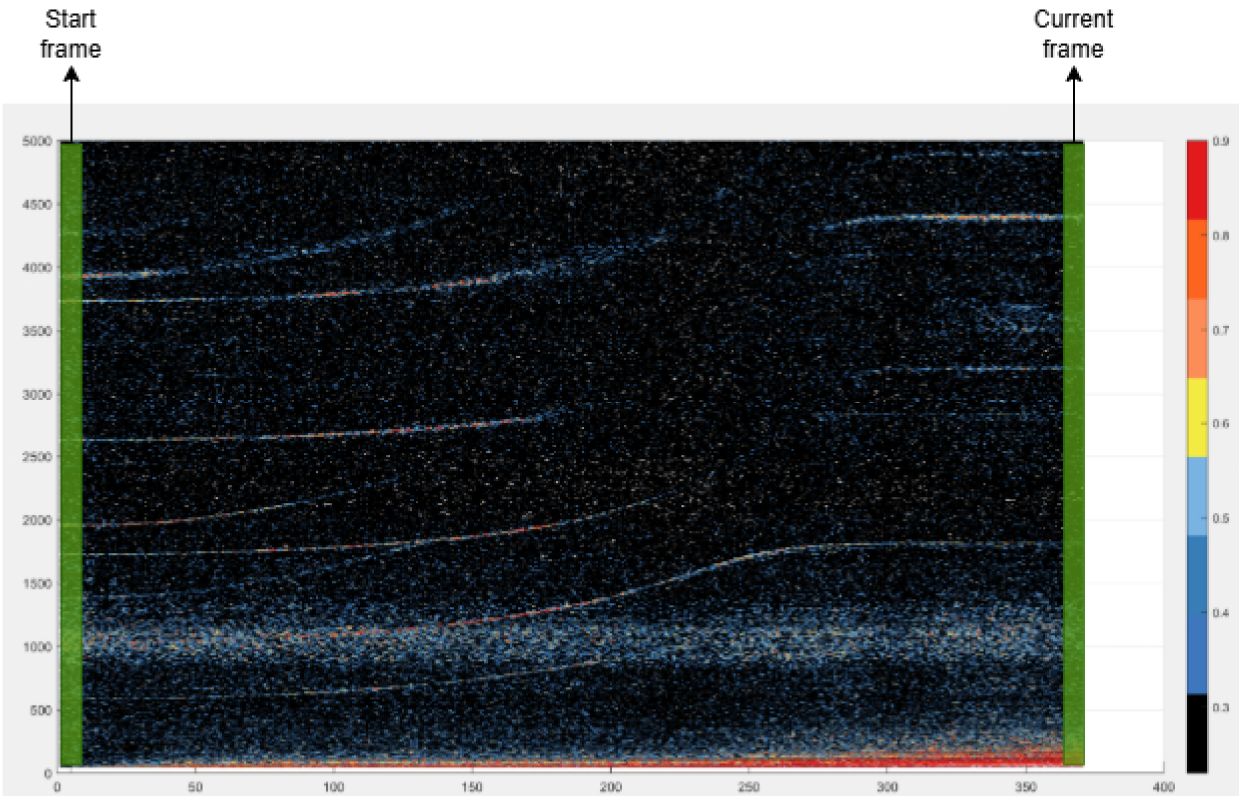

Figure 5

Graph of input frames at the beginning and current points in a sequence of PSDs representing a drain on MPG-FOSS.

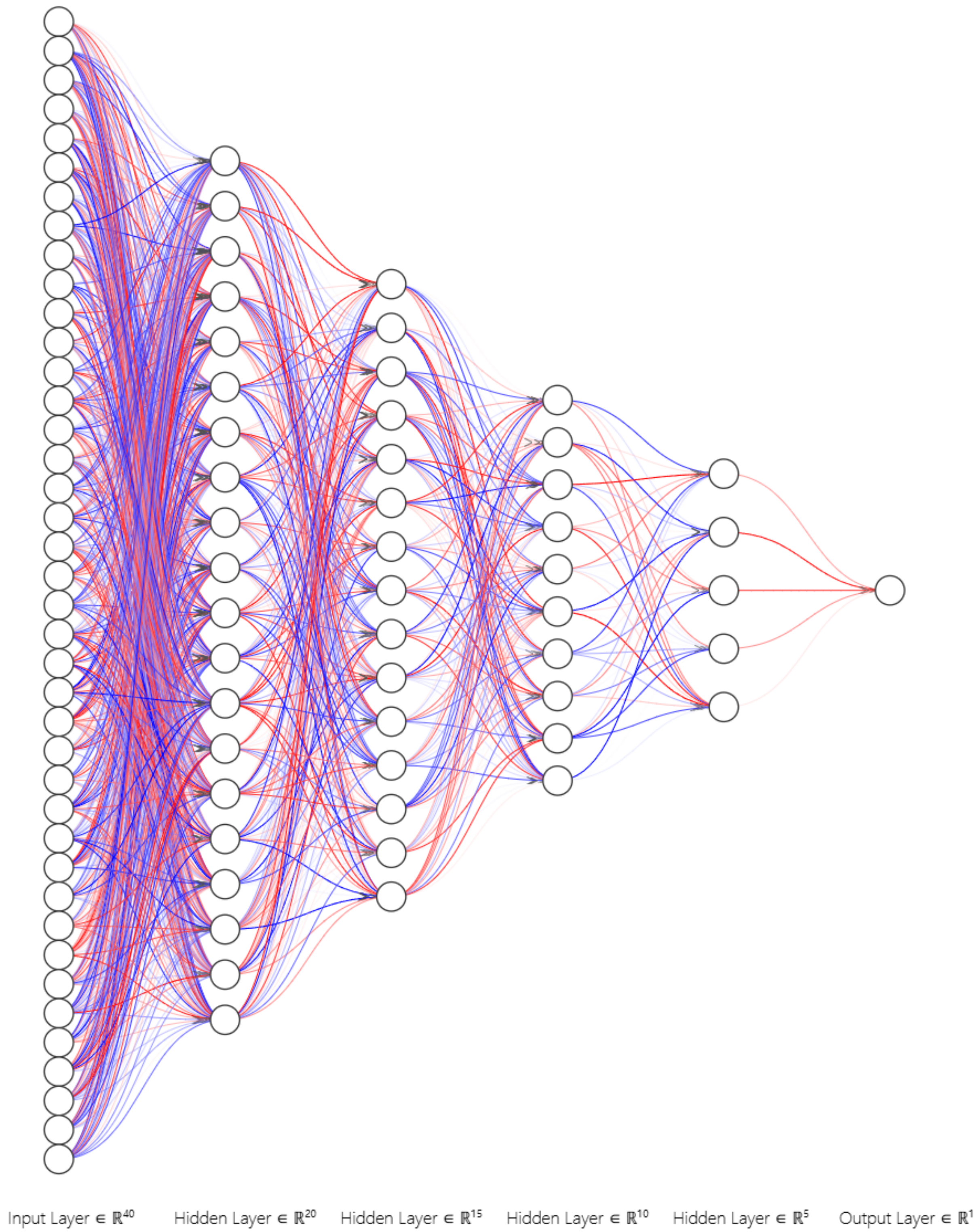

Figure 6

Diagram of the Top-n model architecture where $n = 10$ - red and blue represent higher and lower weights, respectively

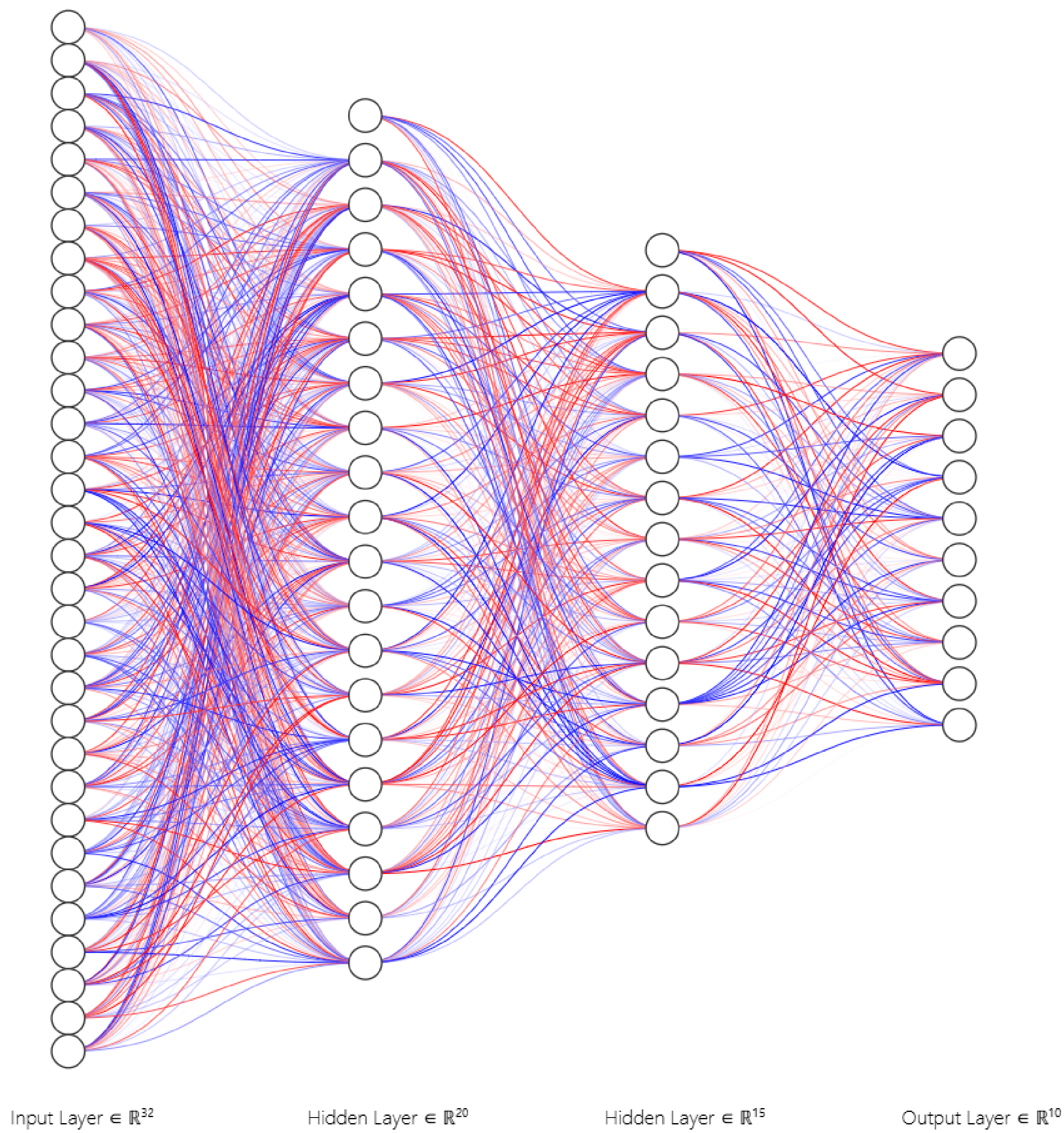

Figure 7

Diagram of the encoder for the Auto-encoder-based model architecture - which takes a complete PSD as input and produces a set of 10 *values as output. This is fed to the model depicted in Fig. 8.*

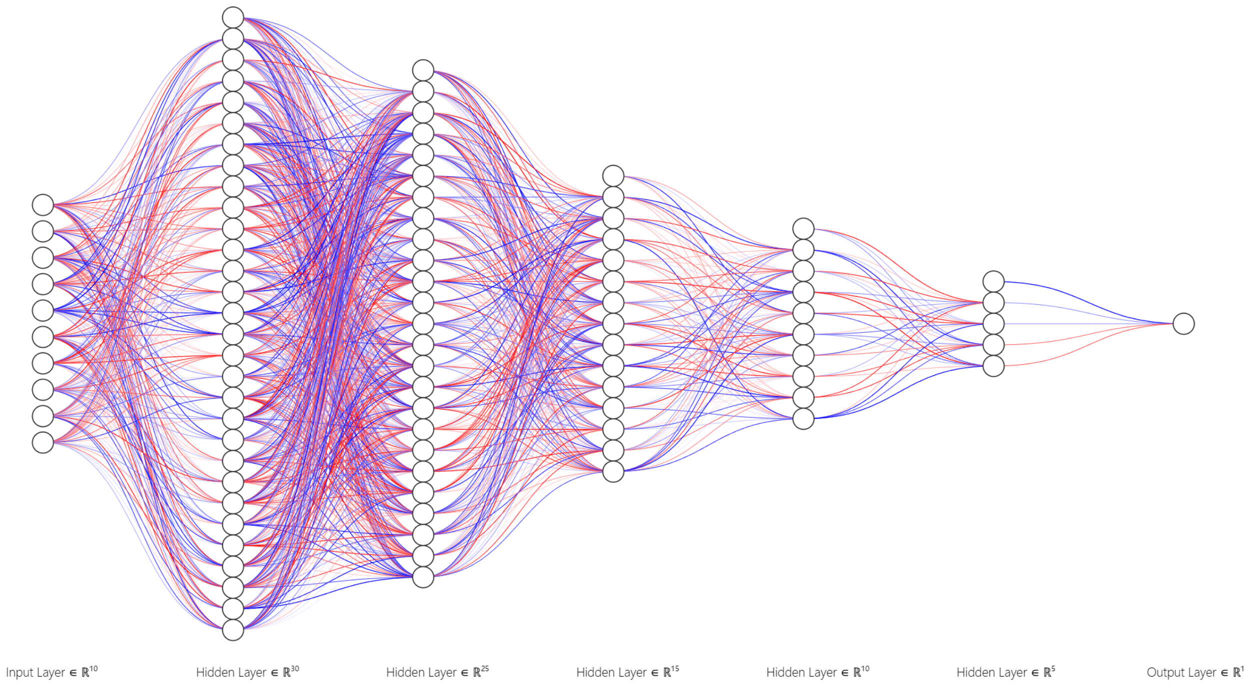

Figure 8

Diagram of the Auto-encoder-based model - which takes an encoded dimensionally-reduced version of the input PSD and outputs a fill level.