Gradient Enhanced Dense Neural Network in Design of Aircraft Wings

Nicolas Dhers (nicolas.dhers@mail.mcgill.ca) Supervisor: Siva Nadarajah
Computational Aerodynamics Laboratory, McGill University, Canada

SUMMER UNDERGRADUATE RESEARCH IN ENGINEERING

Motivation

- Simulation based design and optimization have gained tremendous recognition in the aerospace industry.
- The use of high-fidelity approaches have typically been demonstrated through gradient-based optimization techniques, but these approaches often require extensive computational resources.
- Paired with high fidelity solvers, data-driven surrogates have the potential to produce less expensive approximations and predictions.

Objectives

- The objective of this summer research project is to revisit this research problem and determine the optimal neural network structure and model to be coupled with the in-house computational aerodynamics analysis and design code.
- The code would predict Lift and Drag coefficients as well as their partial derivative with respect to spatial dimensions based on airfoil geometry and flow conditions. The in-house numerical code as well as Xfoil code will be used to populate the needed datasets.

Acknowledgements

Thank you to Prof. Nadarajah for his guidance and support throughout this summer research. Thank you to Mr. Rubin Gruber for his financial support.

References

- Dupuis, R. (2019). Surrogate Models Coupled with Machine Learning to Approximate Complex Physical Phenomena Involving Aerodynamic and Aerothermal Simulations (Doctoral dissertation).
- Giannakoglou, K. C., Papadimitriou, D. I., & Kampolis, I. C. (2006). Aerodynamic shape design using evolutionary algorithms and new gradient-assisted metamodels. *Computer methods in applied mechanics and engineering*, 195(44-47), 6312-6329.

Methodology

- From Multi-Layer Perceptron (MLP) neural network theory, $h_{l+1,k}$ is the input to node k on layer $l+1$:

$$h_{l+1,k} = \sum_{m=1}^{N_l} w_{k,m} G(h_{l,m})$$

where m is the neuron index of layer l and G its activation function.
- The gradient output of the k^{th} neuron on layer $l+1$:

$$v_{l+1,k}^{(p)} = G'(h_{l+1,k}) \sum_{m=1}^{Nl} w_{k,m} v_{l,m}^{(p)}$$

where $v_{l+1,k}^{(p)}$ denotes the derivative of output $v_{l+1,k}$ with respect to the p^{th} component of input vector x.
- The local training error at the n^{th} training epoch for the t^{th} training pattern, where o and ζ are the predicted output and numerical solutions respectively, is then given by:

$$E_t^n = \frac{1}{2}\left(\zeta^{(t)} - o^{(t),n}\right)^2 + \frac{1}{2}\sum_{p=1}^{N}\left(\frac{\partial \zeta^{(t)}}{\partial x_p} - \frac{\partial o^{(t),n}}{\partial x_p}\right)^2$$

- Weights are updated using:

$$w_{k,m}^{n+1} = w_{k,m}^{n} - \eta \frac{\partial E^n}{\partial w_{k,m}}$$

where η is the network's learning rate and:

$$\frac{\partial E^n}{\partial w_{k,m}} = \Delta w_{k,m} = \delta_{l,k}^0 v_{l-1,m} + \sum_{p=1}^{Nl}\left(\delta_{l,k,p}^1 v_{l-1,m} + \delta_{l,k,p}^1 v_{l-1,m}^{(p)}\right)$$

where $\delta_{l,k}^i$ $(i = 0,1,2)$ are calculated recursively:

$$\delta_{l,k}^0 = \sum_{j=1}^{N_{l+1}} w_{j,k}\delta_{l+1,j}^0 G'(h_{l,k})$$

$$\delta_{l,k,p}^1 = \sum_{j=1}^{N_{l+1}} w_{j,k}\left(\delta_{l+1,j,p}^1 G'(h_{l,k}) + \delta_{l+1,j,p}^2 G''(h_{l,k}) h_{l,k}^{(p)}\right)$$

$$\delta_{l,k,p}^2 = \sum_{j=1}^{N_{l+1}} w_{j,k}\delta_{l+1,j,p}^2 G'(h_{l,k})$$

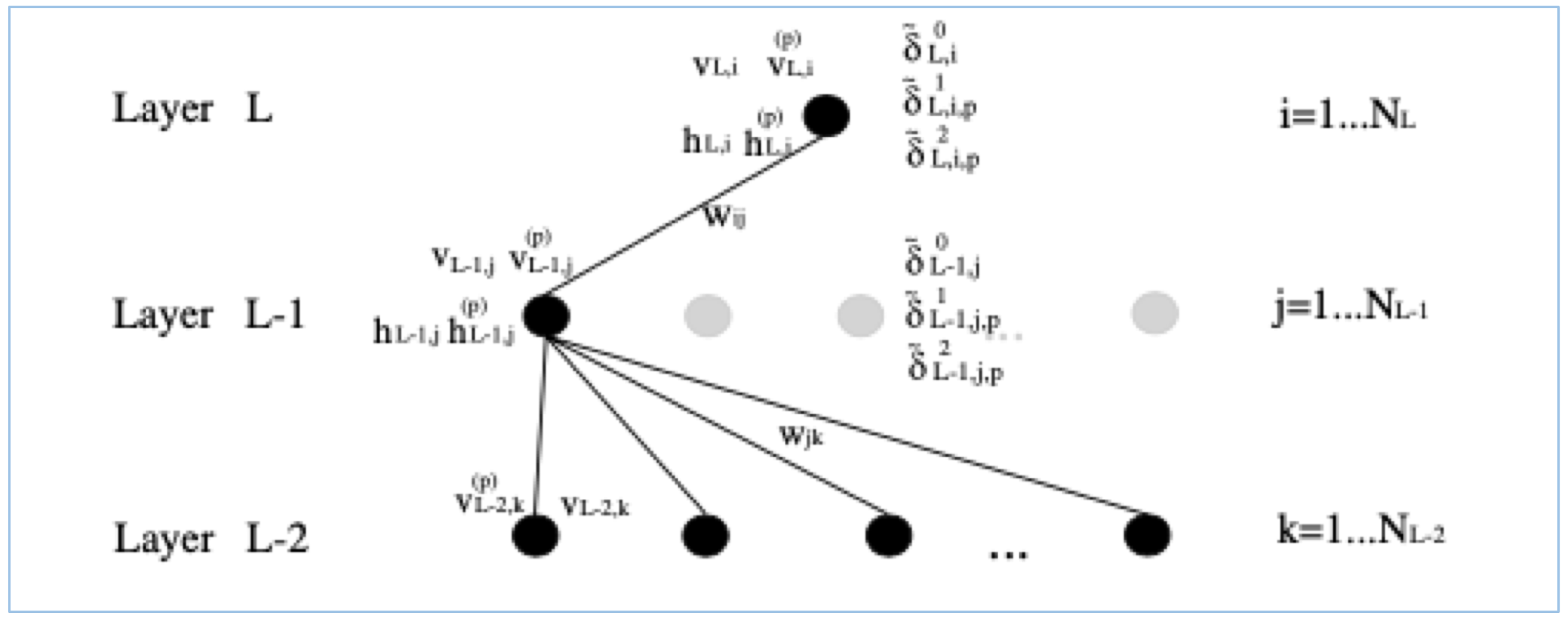

Figure 1: Gradient enhanced network layers illustration

Results

- Epoch error convergence during the training phase was compared for a non-enhanced MLP neural network and a gradient enhanced MLP neural network with similar architectures.
- Figure 2 shows the log of Mean Squared Error (MSE) of both networks converging to a plateau due to the limited size of the training sample.
- Gradient enhanced network's error rate of convergence is much greater

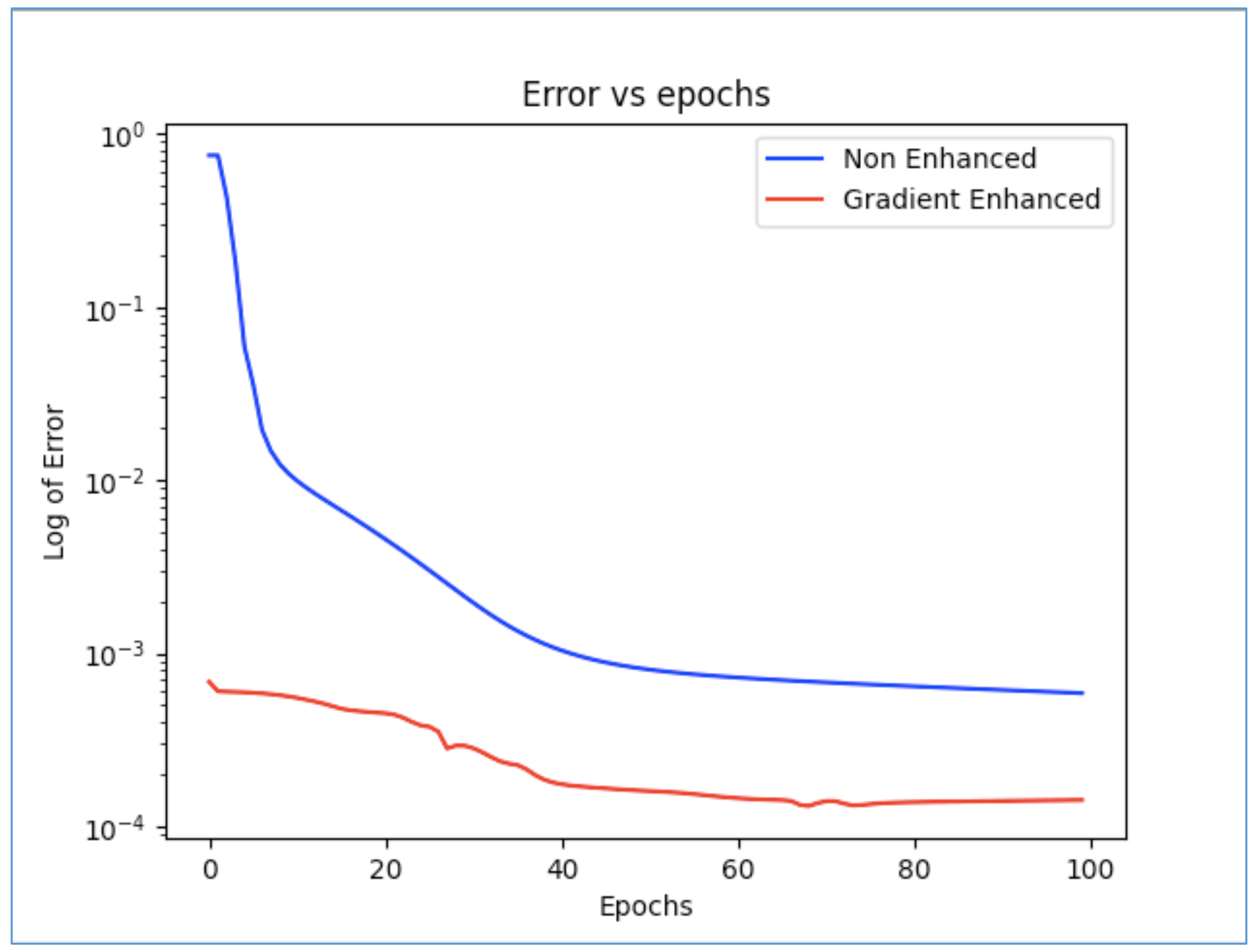

Figure 2: Log(error) versus number of training epochs for both networks

Conclusion

- Gradient enhanced neural network takes full advantage of the in-house computational code by utilizing both aerodynamic coefficients and their derivatives to reach error minima much faster than standard MLP neural networks, hence turning out to be an invaluable tool to cut computational time in wing performance optimization.
- Further work to be done in order to explore the performance of gradient enhanced neural networks with different hyperparameters and architectures paired with computational aerodynamic codes.