A deep learning perspective of the forward and inverse problems in exploration geophysics

Jian Sun*, Zhan Niu, Kris Innanen, Junxiao Li, Daniel Trad

sun1@ucalgary.ca

Abstract

Deep learning becomes to be a very powerful and efficient technique in many fields, where the recurrent neural network (RNN) has significant benefits of exhibiting temporal dynamic behavior for time dependency tasks by building a directed graph of a sequence. In this paper, with a self-designed RNN framework, the forward modeling of wave propagation is casted into a forward propagation of RNN, which allows the inversion problem being treated as the training process of RNN. Using this specific network, we numerically analyze the influence and playing role of learning rate (i.e., step-size) for each gradient-based optimization algorithm. Comparisons of gradient-based and non-linear algorithms are also discussed and analyzed. To examine our analysis, the Marmousi model is employed to perform the inversion on the proposed RNN using both gradient-based and non-linear algorithms.

The forward problems in deep learning framework

The wave equation for an acoustic medium with constant density in time domain:

$$\nabla^2 \mathbf{u}(\mathbf{r},t) = \frac{1}{v^2(\mathbf{r})}\frac{\partial^2 \mathbf{u}(\mathbf{r},t)}{\partial t^2} + \boldsymbol{s}(\mathbf{r},t)\delta(\mathbf{r}-\mathbf{r}_s) \quad (1)$$

where, ∇^2 denotes the spatial-Laplacian operator. $\mathbf{u}$ represents pressure or displacement. The source term is denoted by $\mathbf{s}$. Using the second-order finite-difference method, the mathematical formulation for wavefield is written as,

$$\mathbf{u}(\mathbf{r},t+\Delta t) = v^2(\mathbf{r})\Delta t^2\left[\nabla^2\mathbf{u}(\mathbf{r},t) - \boldsymbol{s}(\mathbf{r},t)\delta(\mathbf{r}-\mathbf{r}_s)\right] + 2\mathbf{u}(\mathbf{r},t) - \mathbf{u}(\mathbf{r},t-\Delta t) \quad (2)$$

Equation 2 shows that the forward modeling of wave propagation can be considered as an iterative process which takes the source term and wavefields at the two previous time steps as inputs.

Instead of the traditional way of coding forward problem, we cast it into a RNN framework whose unrolled graph is shown in Figure 1, and the single cell is plotted in Figure2.

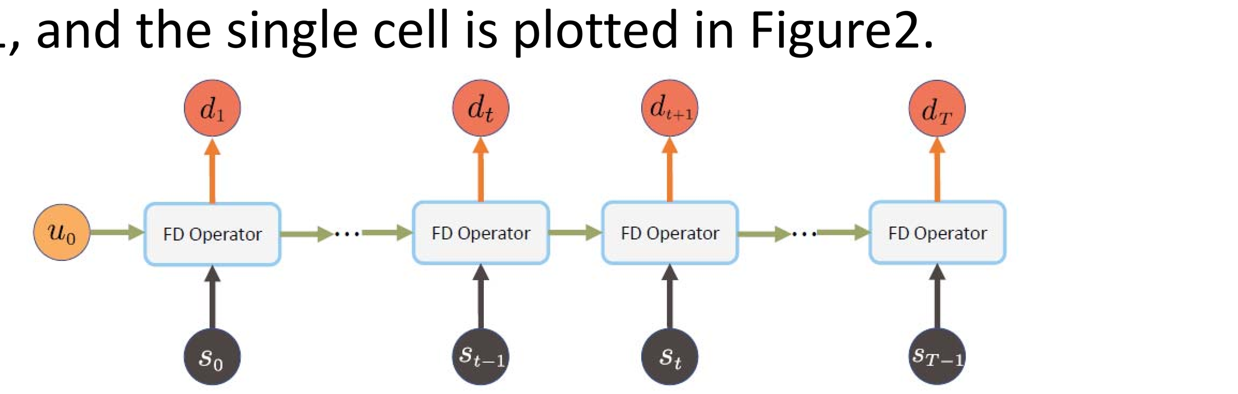

FIG. 1. The unrolled directed acyclic graph of RNN for the forward problem.

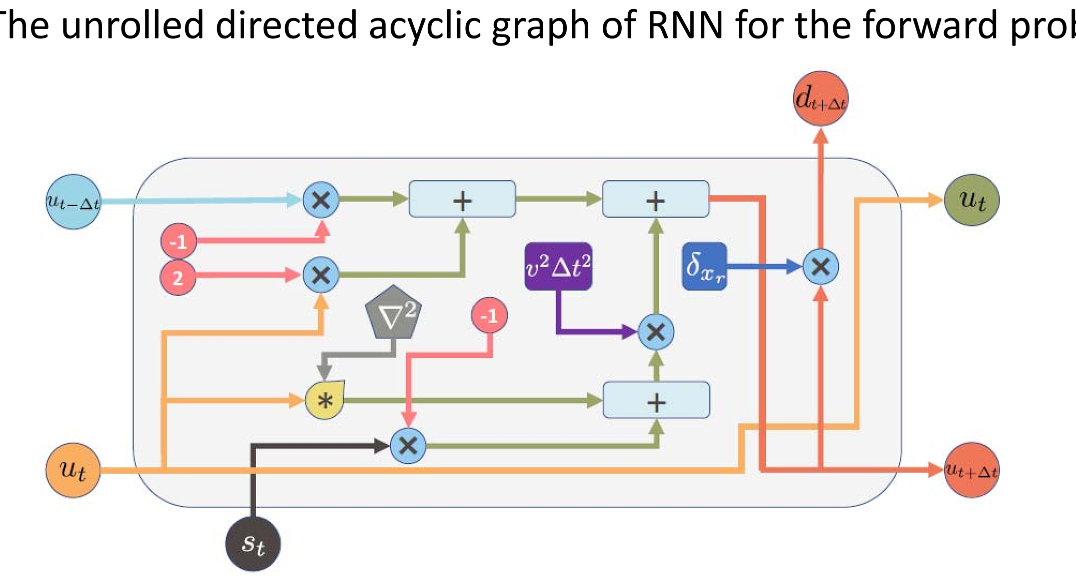

FIG. 2. The single cell's architecture of RNN for the forward problem.

The inverse problems in deep learning framework

The mathematical formulation of the least-square norm of the misfit function is written as,

$$J(v) = \frac{1}{2n_s}\sum_{\mathbf{r}_s}\sum_{\mathbf{r}_g}\sum_{t}(\mathbf{d}_t - \tilde{\mathbf{d}}_t)^2 \quad (3)$$

At n-th iteration, the model is updated through a perturbation,

$$v_{n+1}(\mathbf{r}) = v_n(\mathbf{r}) + \alpha\delta v_n(\mathbf{r}) \quad (4)$$

where α represents the step-size at the current iteration, which is usually being called as learning rate in deep learning. Using gradient-based algorithm, the perturbation is achieved by negative gradient,

$$\delta v_n(\mathbf{r}) = -g_n(\mathbf{r}) \quad (5)$$

In perspective of RNN's framework, the gradient can be derived using chain rule and written as,

$$\begin{aligned} g &= \frac{\partial J}{\partial v} \\ &= BP\Big(-\frac{1}{n_s\boldsymbol{v}^2\Delta t^2}\sum_{\mathbf{r}_s}\sum_{\mathbf{r}_g}\boldsymbol{\delta}\mathbf{d}_t\Big)\frac{\partial\tilde{\boldsymbol{u}}_t}{\partial\boldsymbol{v}} \\ &\approx BP\Big(-\frac{1}{n_s}\sum_{\mathbf{r}_s}\sum_{\mathbf{r}_g}\boldsymbol{\delta}\mathbf{d}_t\Big)\frac{2}{\boldsymbol{v}^3}\frac{\partial^2\tilde{\boldsymbol{u}}_t}{\partial t^2} \end{aligned} \quad (6)$$

Numerical analysis of hyperparameter selection

To numerically analyze the hyperparameter effects and speed of convergence, gradient-based (including GD, Momentum, Adagrad, RMSprop, and Adam) and non-linear optimization algorithms (including CG, L-BFGS, and TNC) are implemented on a depth-varying velocity profile (Figure 3).

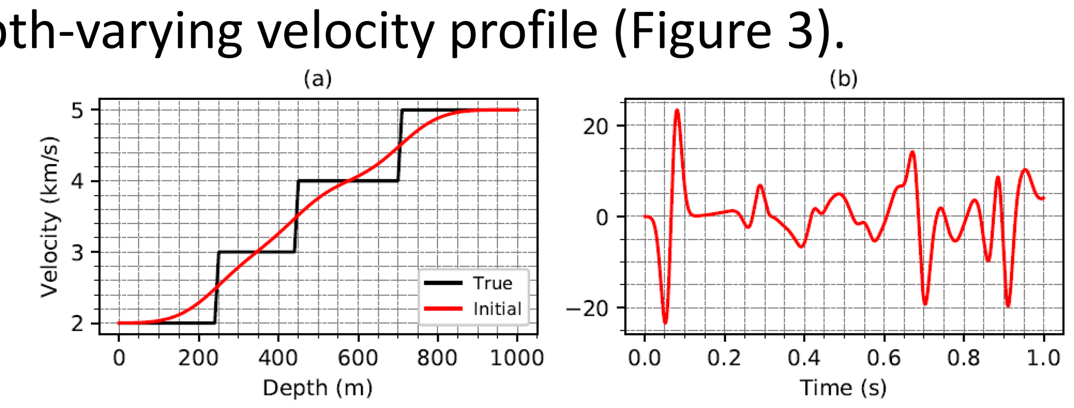

FIG. 3. A depth-varying model. (a) Black: True. Red: Initial. (b) Synthetic data.

The learning rate tuning for each gradient-based algorithm is discussed and theoretically analyzed in the reports. Their best performances with appropriate learning rate are plotted in Figure 4. Comparison of Adam, CG and L-BFGS is shown in Figure 5.

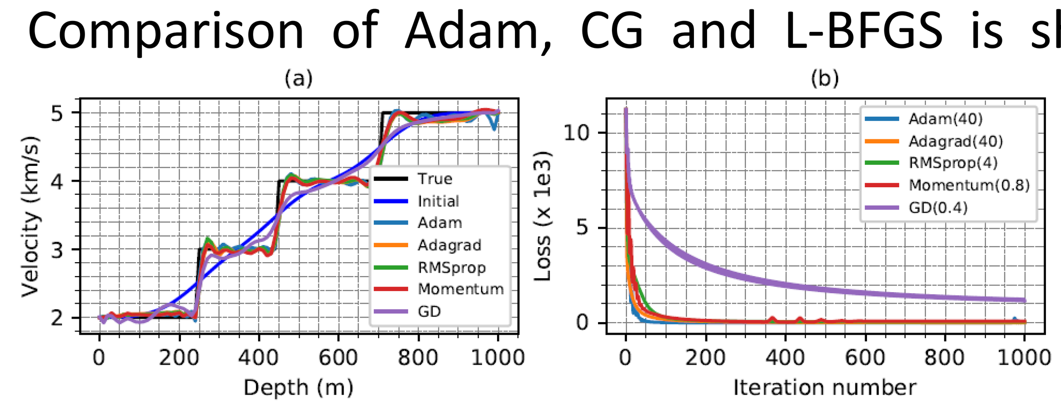

FIG. 4. Comparison of gradient-based algorithms. (a) Final inversion results. (b) Convergence of objective function versus iteration.

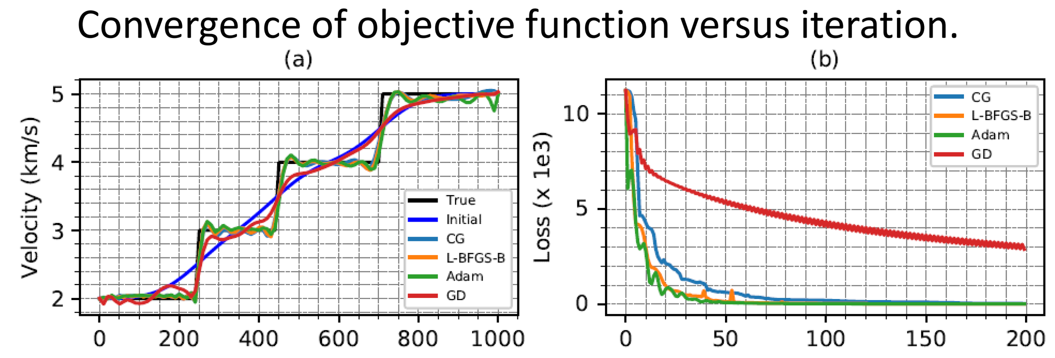

FIG. 5. Comparison of Adam, CG, and L-BFGS. (a) Inversion. (b) Convergence.

Synthetic example of Marmousi

The 2D Marmousi model is employed to fully examine the capacity of RNN framework for inversion problem. 12 shots are emitted using 12Hz of Ricker wavelet. With same initial model, inversion using Adam, CG, L-BFGS are shown in Figure 6, 7, 8, respectively. The comparison of convergence for objective function is plotted in Figure 9.

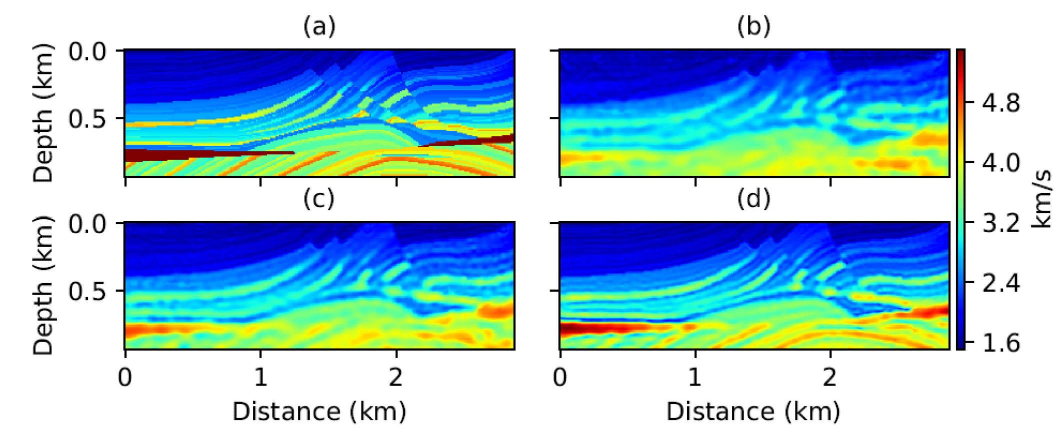

FIG. 6. Inversion with Adam. (a) True. (b) 25th. (c) 50th. (d) 100th iteration.

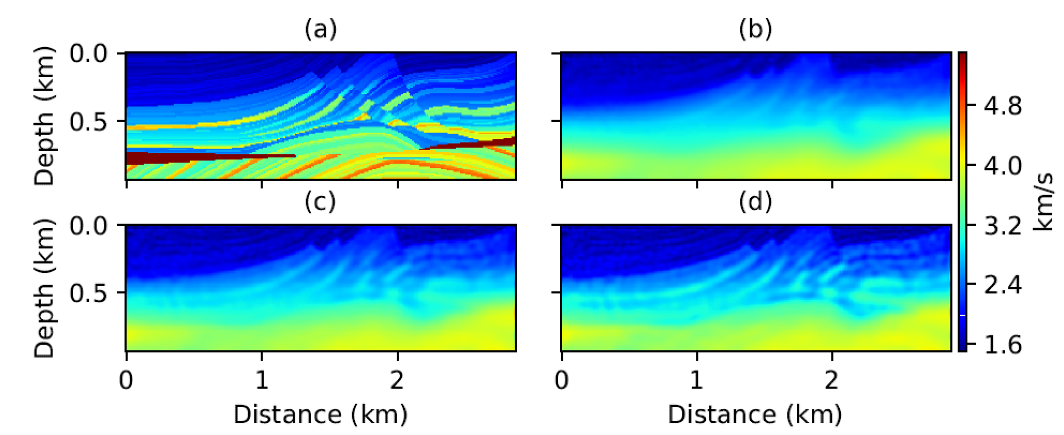

FIG. 7. Inversion with CG. (a) True. (b) 400th. (c) 800th. (d) 1420th iteration.

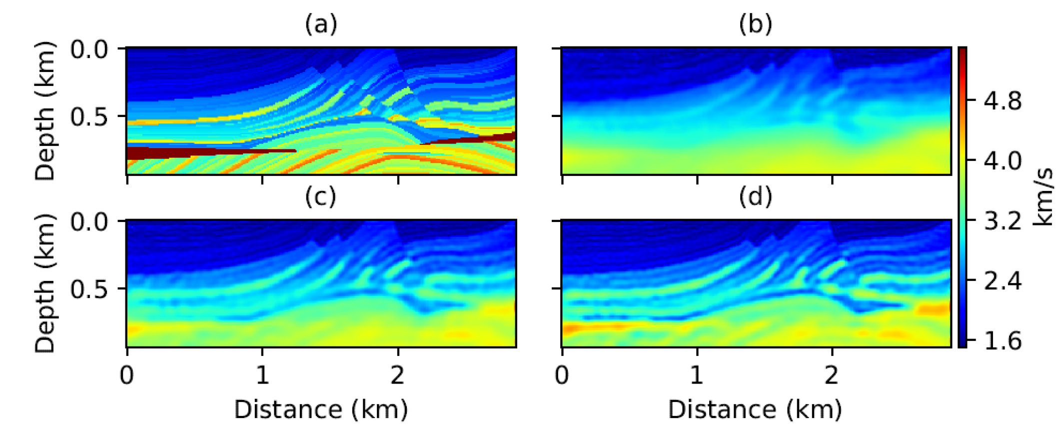

FIG. 8. Inversion with L-BFGS. (a) True. (b) 200th. (c) 600th. (d) 1000th iteration.

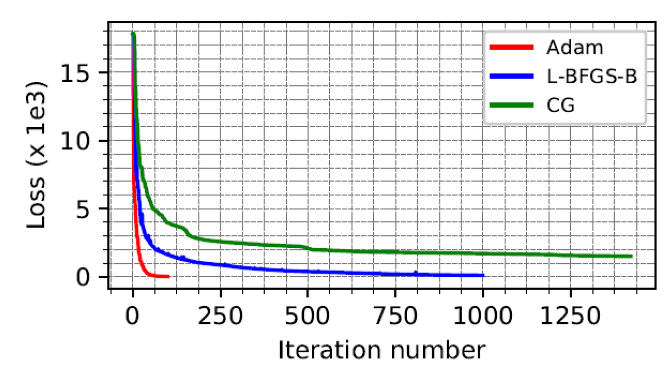

FIG. 9. Comparison of convergence using Adam, non-linear CG, and L-BFGS.

Conclusion

We implement inversion in a RNN framework which uses full wavefield information. The gradient derived in a deep learning perspective is equivalent to FWI. By numerically analyzing the effects of hyperparameter tuning, we recommend the best appropriate learning rate range for variant gradient-based algorithms on geophysical velocity inversion problem. Comparing to non-linear CG and L-BFGS algorithms, Adam shows competitive advantages and faster convergence speed of objective function, which is examined and proven by the 2D Marmousi model implementation.

Acknowledgements

We thank the sponsors of CREWES for continued support.

CREWES

www.crewes.org

NSERC CRSNG

UNIVERSITY OF CALGARY
FACULTY OF SCIENCE
Department of Geoscience