Sensitivity Study of Mini-Batch Size on a Long Short-Term Memory Network for In-situ Sensing of Core-to-shell Ratio of Microencapsulated Phase Change Materials

Rebecca C. Shannon
Department of Mechanical Engineering
Western New England University
Springfield, United States
rebecca.shannon@wne.edu

Jingzhou Zhao
Department of Mechanical Engineering
Western New England University
Springfield, United States
jingzhou.zhao@wne.edu

Zhaojun Li
Department of Industrial Engineering and Engineering Management
Western New England University
Springfield, United States
zhaojun.li@wne.edu

Jingru Benner
Department of Mechanical Engineering
Western New England University
Springfield, United States
jingru.benner@wne.edu

***Abstract*—Microencapsulated phase change materials are being studied for applications for thermal energy storage in concentrated solar fields. During fabrication, the thickness of the encapsulation cannot be readily measured for real-time control. Therefore, a machine learning network, specifically a Long Short-Term Memory network, is being developed to estimate the ratio of the shell radius to core radius based on a one second temperature history. The mini-batch size determines how often the algorithm weights are updated during network training, and shuffle indicates whether the training data is shuffled during training. A general factorial design is used to analyze the effects of varying mini-batch size and shuffle, along with the core-to-shell ratio, on the RMSE of the response from the Long Short-Term Memory network. It was found that the network performed better for smaller core to shell ratios (less than 0.6) and had the lowest RMSE when the mini-batch size was 128. The minimum RMSE found was 0.00501.**

Keywords—long short-term memory, machine learning, mini-batch size, phase change materials

I. Introduction

Phase change materials (PCMs) are those that can store energy in their latent heat of phase change, and they are being studied for applications in energy storage for concentrated solar power. While organic and eutectic salt PCMs are widely used from low to medium temperature applications, desirable materials for future high temperature PCMs are metals like copper and aluminum for their high volumetric energy density and high thermal conductivity. A microencapsulated phase change material (MEPCM) is a small particle of PCM (<50 μm radius) with an encapsulating shell (thickness <50 μm). A schematic of a MEPCM can be seen in Fig. 1. In Fig. 1a, the PCM core is in the solid phase; to store energy, the particle's temperature is raised until the core changes to the liquid phase (Fig. 1b). To release the energy, the temperature of the particle is lowered and the core returns to the solid phase and releases its latent heat. The shell remains in the solid phase for the entire process to contain the core material [1].

To ensure the performance of the MEPCM, it is desirable to monitor the shell thickness during fabrication. However, the direct non-destructive measurement of shell thickness on a MEPCM particle requires the use of High-speed Synchrotron X-ray Imaging to distinguish the shell from the core at high frame rates, which is only available for research instead of production environment. An in-situ sensing method which makes use of the temperature history during cooling of the MEPCM to indirectly estimate the core-to-shell ratio is therefore proposed. The temperature history can be readily captured using a high-speed thermal camera [2].

To study the fundamental feasibility of the proposed method, simulated temperature histories of MEPCM during cooling are being used as model inputs [3]. Estimating the shell thickness of the PCM is expected to be accomplished using the machine learning program Long Short-Term Memory (LSTM). LSTM is a recurrent neural network (RNN) that is better equipped to handle larger gaps between the supplied information and the necessary response. In a traditional RNN, the error signals tend to either blow up or vanish when they have to travel "back in time" for information. To remedy this, Hochreiter and Schmidhuber [4] introduced LSTM, which utilizes constant

Funded by the Northeast Cyberteam at the Massachusetts Green High Performance Computing Center under the National Science Foundation Award No. OAC-1659477.

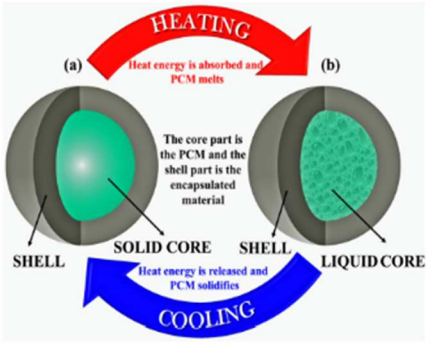

Fig. 1. Working principle of MEPCM [1].

error flow. This network consists of a memory cell with several gates. The memory cell is where information is stored for a given timestep. The input gate determines what incoming information should be saved and protects the cell from unnecessary information, and the output gate decides which information currently in the cell's memory should be passed on to the next timestep. In other words, what new information needs to be remembered and what current information can be forgotten by the cell. The cell is updated to reflect the information from these gates, and the cycle begins again with the new memory cell [4].

In LSTM, the training data sets are used to train the network to predict the desired results, and later the testing data is used to test the accuracy of the network. While the network is being trained, the weights for the algorithms are continuously being updated in order to create the most accurate network. The mini-batch size is used to define how often these updates take place. In batch training, the algorithms are updated only after all of the training data sets have been considered. Opposite batch training is the stochastic method, in which the algorithm is updated after each individual data set is considered, which is when the algorithm is updated most frequently. Mini-batches fall between these two methods by looking at a portion of the training data set before updating the algorithm weights. The mini-batch size is typically a power of 2. Smaller mini-batch sizes (and the stochastic method) are more computationally expensive because they take more time [5].

Within the LSTM framework, the shuffle option goes hand in hand with the mini-batch size, and it determines if and when the training data is shuffled. When the number of training data sets is not evenly divisible by the mini-batch size, the extra data sets are thrown out. If shuffle is set to never, the training data appears in the same order each epoch, and the same data sets are discarded each epoch. When set to every epoch, the training data is consistently shuffled so that different data sets are thrown out each epoch [6].

In this research, a sensitivity study is performed to learn the effect of the mini-batch size on the response of an LSTM network used to estimate the geometric properties of an MEPCM. Additionally, the effects of the shuffle option are also studied. Because the testing data sets have different core to shell ratios, this factor is also considered in the study.

II. Methods

A. Data Collection

In this study, two set of data were used: first cooling curves to be used in the machine learning network and second results from different iterations of the network. The first set of data was collected using an OpenFOAM model of a MEPCM particle. This model returns the temperature of the particle center, surface, and core-shell interface for every millisecond over a one second time interval [3]. The temperature histories of the center and the surface of the particle were used, and the particle had a copper core with a fused silica shell. The core radius, core to shell ratio, ambient temperature, and convective heat transfer coefficient were varied to simulate different sized particles as well as different operating conditions, shown in Table 1. The

TABLE I. Properties of Studied MEPCM

Factor Information		
Variable	***Values***	***Units***
Core Radius, r_{core}	10, 20, 30, 40, 50	µm
Core to shell Ratio, φ	0.2, 0.4, 0.6, 0.8, 1.0	n/a
Ambient Temperature, T_{HF}	293, 298	K
Convective Heat Transfer Coefficient, h	100, 200	W/[m^2·K]

core to shell ratio is defined as $\phi = r_{core}/t_{shell}$, with r_{core} and t_{shell} being the core radius and the thickness of the shell, respectively. The initial temperature of the particle was assumed to be 2,000 K, and an example cooling curve is shown in Fig. 2. A total of 100 data sets were collected.

The second set of data was collected using an LSTM network in MATLAB. The network has one LSTM layer with 500 hidden units. A dropout layer of 0.1 is used for 1,000 epochs. The LSTM layer is a regression layer, so it takes the sequence input data and returns a single value, the core to shell ratio, which is being estimated. The mini-batch size in the network training was varied to determine its effects on the results from the network. Additionally, for each mini-batch size, two cases were run: one with the data being shuffled every epoch and the other with it never being shuffled. Eighty-five percent of the data collected from OpenFOAM was used to train the network while the remaining 15% was used to test the network. All five possible core to shell ratios were represented in the testing data. To train the network, the temperature vs time data from the center and the surface of the MEPCM along with an associated core to shell ratio were used. All of the data was normalized. When testing the network, the temperature data was the input, and the response was the ratio. For each test, the network was run 10 times and the average response over the 10 loops was given as the mean response. Additionally, the root mean square error (RMSE) was calculated over the ten loops. A smaller RMSE indicated a more accurate prediction. There were a total of 15 test cases.

B. General Factorial Design

A general factorial design (Table 2) was used to analyze the

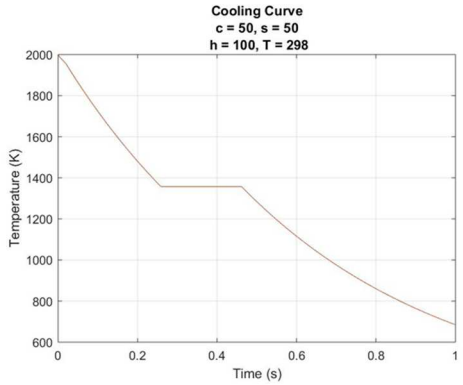

Fig. 2. Example cooling curve, where c, s, h, and T are the core radius, shell thickness, heat transfer coefficient, and ambient temperature, respectively.

TABLE II. GENERAL FACTORIAL DESIGN

Factor Information		
Factor	***Levels***	***Values***
Mini-Batch Size	7	2, 4, 8, 16, 32, 64, 128
Shuffle	2	Every Epoch, Never
Ratio	5	0.2, 0.4, 0.6, 0.8, 1.0

effects of the mini-batch size and the shuffle setting on the RMSE results from the LSTM network. The levels for the mini-batch size were selected as powers of 2^k with $1 \leq k \leq 7$, and the shuffle option was set to every epoch or never. Additionally, the core to shell ratio was included as a factor. The general factorial design was used to evaluate which factors and interactions have a significant effect on the results, and this information can be used to optimize the network. Minitab was used to analyze this data.

III. RESULTS

When the general factorial design was run using 5% significance, the analysis of variance in Table 3 showed that the ratio and shuffle factors were significant, and the interaction between these two factors was also significant ($P < 0.05$). The main effects plot in Fig. 3 indicates that when shuffle is set to every epoch, the RMSE is reduced by 43.7% compared to when the data is never shuffled. Additionally, the RMSE was much lower for $\phi = 0.2$, 0.4, and 0.6. The highest RMSE for each of these ratios was 0.01121, while the smallest value for the larger ratios was 0.02933. This indicates that the network is biased towards the cases with a smaller core to shell ratio. The mini-batch size of 128 resulted in the lowest RMSE (0.01034). The overall mean RMSE for this design was 0.02016.

It is also worth noting that from Fig. 4, when the mini-batch size was 128, the RMSE was similar whether shuffle was set to never or every epoch. The mini-batch size of 128 represents batch training because it is greater than the 85 training data sets. Since all of the training data sets are considered before the network algorithms are updated, the order that the data sets are

TABLE III. ANOVA TABLE FOR FIRST GENERAL FACTORIAL DESIGN

Analysis of Variance					
Source	***DF***	***Adj SS***	***Adj MS***	***F***	***p-value***
Mini-Batch Size	6	0.004289	0.000715	0.66	0.680
Shuffle	1	0.006670	0.006670	6.18	0.014
Ratio	4	0.039919	0.009980	9.25	0.000
Mini-Batch Size*Shuffle	6	0.001908	0.000318	0.29	0.939
Mini-Batch Size*Ratio	24	0.006477	0.000270	0.25	1.000
Shuffle*Ratio	4	0.013966	0.003491	3.23	0.014
Mini-Batch Size*Shuffle*Ratio	24	0.007057	0.000294	0.27	1.000
Error	140	0.151121	0.001079		
Total	209	0.231406			

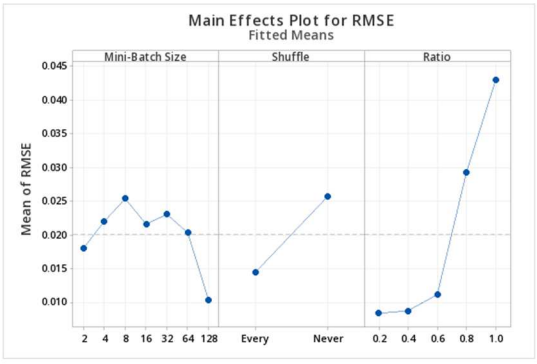

Fig. 3. Main effects plot for first general factorial design.

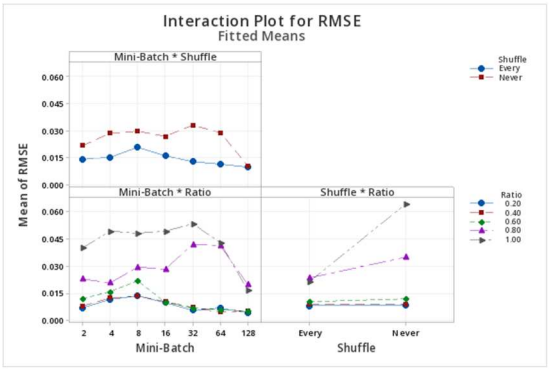

Fig. 4. Interaction plot for first general factorial design.

considered (determined by the shuffle setting) is not as important as it is in the mini-batch training. Fig. 4 also graphically shows that the mini-batch size to shuffle interaction and the mini-batch size to ratio interaction were not significant because each of the curves on these plots follow similar trends. The interaction between shuffle and the ratio can be seen to be more significant because the curves are not parallel and intersect each other.

Looking at the residual plots in Fig. 5, the residuals do not seem to closely follow the normal distribution. Furthermore, the versus fits and versus order plots show an increasing trend. These residual plots indicate that the assumptions of normal and

independently distributed error were not followed in the model. The R-Squared value for this study was 34.69%, indicating that the study was not a good fit for the data.

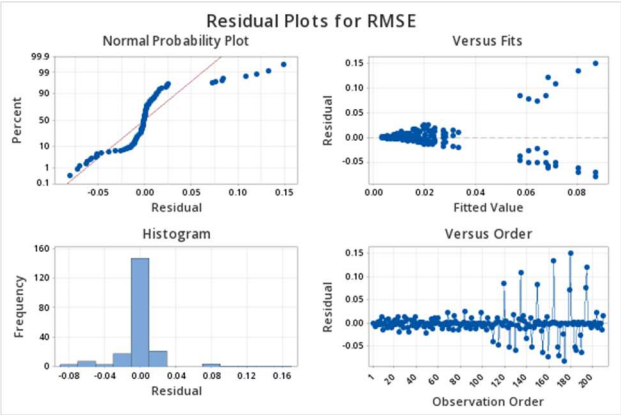

Fig. 5. Residuals plot for first general factorial design.

Because the RMSE was much higher for $\phi = 0.8$ and 1.0, a new general factorial design was done only including $\phi = 0.2$, 0.4, and 0.6. The new ANOVA table (Table 4) indicates that while the ratio is still a significant factor, the mini-batch size had a higher significance (P = 0.018 vs P = 0.000, respectively). With the R-Squared value of 59.55%, the data in the second study was a better fit for the model. The residuals for this study, shown in Fig. 6, follow a normal distribution closely, with an outlier observed on the Histogram at 0.016. The residual versus fits plot shows a slight increasing trend, but both this plot and the residual versus order plot are randomly distributed, indicating that there were no major violations to the assumptions. The overall mean RMSE was 0.009492 in this design, which is 52% lower than the previous design. This further shows the model is more accurate for smaller ratios.

The main effects plots in Fig. 7 indicate that the larger mini-batch sizes (32, 64, and 128) have a lower RMSE, with 128 again showing the lowest RMSE of 0.00501. In this factorial design, the shuffle option was not a significant factor, which can be seen in Fig. 7: the mean RMSE changes from 0.009135 to 0.009849 when shuffle is changed from every epoch to never, a 7.82% difference. This is compared to a 69.5% difference between the largest RMSE of 0.01642 (for a mini-batch size of

TABLE IV. ANOVA TABLE FOR SECOND GENERAL FACTORIAL DESIGN

Analysis of Variance					
Source	***DF***	***Adj SS***	***Adj MS***	***F***	***p-value***
Mini-Batch Size	6	0.001907	0.000318	14.19	0.000
Shuffle	1	0.000016	0.000016	0.72	0.399
Ratio	2	0.000189	0.000095	4.22	0.018
Mini-Batch Size*Shuffle	6	0.000222	0.000037	1.65	0.144
Mini-Batch Size*Ratio	12	0.000268	0.000022	1.00	0.459
Shuffle*Ratio	2	0.000014	0.000007	0.32	0.730
Mini-Batch Size*Shuffle*Ratio	12	0.000154	0.000013	0.57	0.857
Error	84	0.001882	0.000022		
Total	125	0.004653			

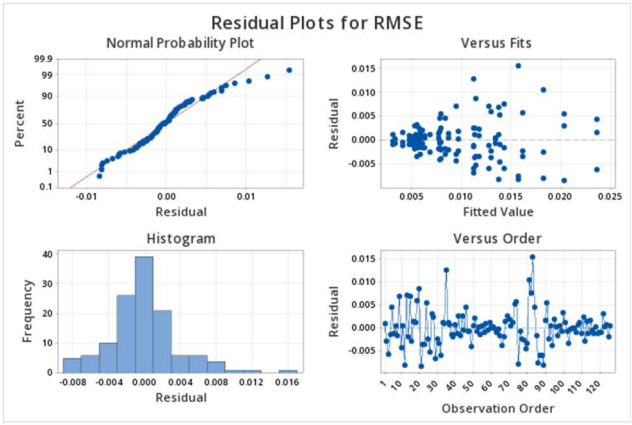

Fig. 6. Residuals plot for second general factorial design.

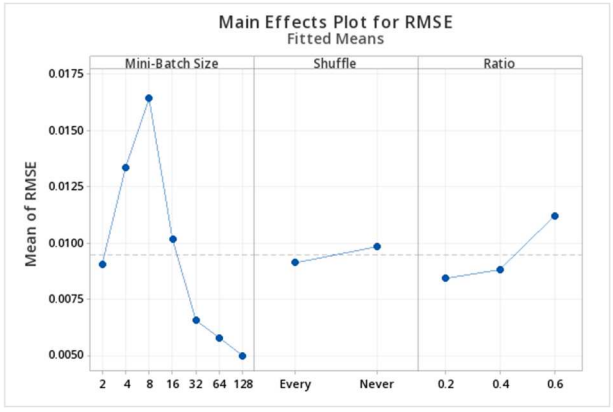

Fig. 7. Main effects plot for second general factorial design.

8) and the smallest RMSE. Similar to the first general factorial design, the cases where the ratio was smaller had a lower RMSE.

IV. CONCLUSION

A general factorial design was used to explore the effects of mini-batch size and data shuffling on the response accuracy of an LSTM network used to estimate the core to shell ratio of MEPCM. The accuracy was measured using the RMSE, which

was calculated over 10 responses for a given set of initial conditions. Because the core to shell ratio varied between 0.2 and 1.0 in steps of 0.2, the ratio was also included as an experimental factor. It was found that the LSTM network had higher accuracy in the test cases with smaller ratios: the ratio was a significant factor and the mean RMSE was larger when the ratio was larger. A second general factorial model was generated using only the test cases where $\phi \leq 0.6$, which increased the fit of the model. In the new design, the ratio and the mini-batch size were considered significant factors. The largest mini-batch size resulted in the smallest RMSE, indicating that the batch training is an optimal setting for this network. Future study will include sensitivity studies for other factors, including the dropout layer and the number of neurons in a hidden layer; expanding the LSTM network to include different materials; and attempting to increase the network's accuracy at higher core to shell ratios.

ACKNOWLEDGMENT

We would like to thank John Goodhue, Julie Ma, and Josh Bevan for computational infrastructure maintenance and support. Additionally, we used computational resources provided by the Northeast Cyberteam project funded by the National Science Foundation at the Massachusetts Green High Performance Computing Center and the Boston University Shared Computing Cluster. This material is based on work supported by the U.S. Department of Energy, office of Energy Efficiency & Renewable Energy under award number DE-EE0009095.

REFERENCES

[1] H. Nazir, et al., "Recent developments in phase change materials for energy storage applications: a review," Int. J. of Heat and Mass Transfer, vol. 129, pp. 491-523, Feb. 2019. [Online] https://doi.org/10.1016/j.ijheatmasstransfer.2018.09.126.

[2] B. Whinery, et al., "Thermal image processing for feature extraction from encapsulated phase change materials," Proc. SPIE 11605, Thirteenth International Conference on Machine Vision, 116050K, 4 January 2021. https://doi.org/10.1117/12.2586979.

[3] J. Benner, et al., "The effect of micro-encapsulation on thermal characteristics of metallic phase change materials," Applied Thermal Engineering, vol. 207, in press https://doi.org/10.1016/j.applthermaleng.2022.118055.

[4] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735-1780, Feb. 1997. https://doi.org/10.1162/neco.1997.9.8.1735.

[5] D. Masters and C. Luschi, "Revisiting small batch training for deep neural networks," 2018. [Online]. Available: https://arxiv.org/abs/1804.07612.

[6] "trainingOptions," MathWorks Help Center, [Online] https://www.mathworks.com/help/deeplearning/ref/trainingoptions.html #responsive_offcanvas