Phase-field modelling of brittle fracture using time-series forecasting*

Minh Ngoc Dinh[1,*], Chien Trung Vo[2], Cuong Tan Nguyen[1], Ngoc Minh La[1]

[1] School of Science, Engineering and Technology RMIT University, Vietnam
[2] Mechanical Engineering and Materials Science University of Pittsburgh, USA

Abstract. The crack propagation behavior can be considered a time-series forecasting problem and can be observed based on the changes of the Phase-field variable. In this work, we study the behavior of the Isotropic Brittle Fracture Model (BFM), and propose a hybrid computational technique that involves a time-series forecasting method for finding results faster when solving variational equations with a fine-grained. We use this case study to compare and contrast two different time-series forecasting approaches: ARIMA, a statistical method, and LSTM, a neural network learning-based method. The study shows both methods come with different strengths and limitations. However, ARIMA method stands out due to its robustness and flexibility, especially when training data is limited because it can exploit a priori knowledge.

Keywords: Brittle Fracture Model · Phase-field · ARIMA · LSTM.

∗ Corresponding author. E-mail address:minh.dinh4@rmit.edu.vn

1 Introduction

Predicting the propagation of cracks that occur on a target material is an important problem in continuum damage mechanics. The BFM [1] is one of the typical models, which bases on the concept of critical energy release rate in order to evaluate and quantify crack formation and propagation. Importantly, the performance of modelling crack formation and propagation is mesh size-dependent, and many have tried to address this issue [2]. Recently, collecting experimental/computational data to predict the crack propagation process using machine learning techniques becomes a mainstream approach.

There are two main approaches in the field of machine learning: statistical methods and artificial neural network (ANN) architectures. Statistical methods such as the Holt-Winters [3] or Autoregressive Intergrated Moving Average (ARIMA) have shown to be robust and flexible, especially when fewer training data is available because they exploit a priori knowledge. However, these techniques often assume the linearity in the data. Besides, ANNs such as Long Short Term Memory (LSTM) [4] can approximate almost any function, given enough training data. In this work, we study the behaviors of the Isotropic BFM. We propose a hybrid framework where we utilize data collected from computational brittle fracture simulations to enhance the time-series forecasting capacity of machine learning techniques, especially when working with fine-grained mesh. We use this case study to compare two different time-series forecasting approaches: ARIMA and LSTM. Our main contributions are (1)

* Supported by RMIT University, Vietnam, Internal Research Grant 2, 2020.

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_36

a hybrid framework involves a time-series forecasting method for predicting crack propagation behaviors, and (2) comparing the performance of traditional forecasting techniques to deep learning-based algorithms.

Section 2 presents the Phase-field variable model. In section 3, we present the framework using ARIMA and LSTM as time-series forecasting techniques. We describe our case study in section 4. We show how time-series forecasting can predict crack propagation behaviors, and evaluate both LSTM and ARIMA models in terms of accuracy and compute performance in section 5. Section 6 presents the conclusion and future work.

2 Isotropic Brittle Fracture Model

2.1 Phase-field approximation

The Phase-field variable $\phi\left(x\right)$ is an approximated exponential function depending on the spatial variable x, which is used to describe the damage state of materials at each position in a continuum domain. The value of $\phi\left(x\right)$ varies in $[0,1]$ as x varies from $(-\infty,0]$ and $[0,+\infty)$ such that for $\phi\left(0\right)=1$ denotes the cracked material and $\phi\left(\pm\infty\right)=0$ represents the intact one. The diffusion of the phase-field variable depends on the length scale parameter l_c, and $\phi\left(x\right)$ is the solution of the homogeneous differential equation, also called the Euler equation of the variational principle: $\phi\left(x\right)-l_c^2\Delta\phi\left(x\right)=0 \ in \ \Omega$
Following the variational approach, the total potential Ψ forms the crack is:

$$\Psi\left(\phi,\mathbf{u}\right)=\int\limits_{\Omega}\left[\left(1-\phi^2\right)+\kappa\right]\psi_0\left(\varepsilon\right)d\Omega+\int\limits_{\Omega}\frac{\mathcal{G}_c}{2}\left[l_0\nabla\phi\cdot\nabla\phi+\frac{1}{l_0}\phi^2\right]d\Omega \quad (1)$$

On the other hand, the total potential energy is equivalent to the internal energy of the system. The variation of the internal energy increment $\delta\mathrm{W_{int}}$ and the external work increment $\delta\mathrm{W_{ext}}$ are similarly expanded as follows:

$$\delta\mathrm{W_{int}}=\delta\Psi=\frac{\partial\Psi}{\partial\varepsilon_{\mathrm{ij}}}\delta\varepsilon_{\mathrm{ij}}+\frac{\partial\Psi}{\partial\phi}\delta\phi;\qquad \delta\mathrm{W_{ext}}=\int\limits_{\Omega}b_j\delta u_j d\Omega+\int\limits_{\partial\Omega}h_j\delta u_j d\partial\Omega \quad (2)$$

The residual $\delta\mathrm{W_{int}}-\delta\mathrm{W_{ext}}$ is a system of equations containing displacement variables and phase-field computed by the principle of virtual work.

2.2 Treating the crack propagation process as a time-series

The crack diffusion of brittle materials, as described by the Phase-field variable, could be fixed in an unchanged-dimensional matrix. Importantly, the brittle fracture model is quasi-static, in which the load steps could be considered as time steps in the process of simulation. Msekh et al. [5] describe the crack propagation as temporal-spatial dependent process. As every mesh node on an arbitrary position has a temporal attribute, we investigate how crack propagation can be predicted using time-series modelling techniques.

3 Time-series forecasting

Time-series forecasting analyses observations to (1) uncover potential structure such as autocorrelation, trend or seasonal variation, and (2) produce monitoring and forecasting capacity. There are two main approaches in time-series forecasting: statistics-based method such as ARIMA and the deep learning-based architectures such as LSTM [4]. This section demonstrates how both techniques can be applied to forecast the crack propagation process.

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_36

3.1 Statistics based time-series forecasting with ARIMA

Assuming time is a discrete variable and X_t denotes the observation at time t, and ϵ_t denotes the zero-mean random noise term at time t. The autoregressive (AR) process, and moving average (MA) can be combined as:

$$X_t = \sum_{i=1}^{k} \alpha_i X_{t-i} + \sum_{i=1}^{q} \beta_i \epsilon_{t-i} + \epsilon_t \quad (3)$$

The combination of the two schemes [6] takes both previous observations and the progressive noise term into account: the "autoregressive" term k: presents the lags of the series in the forecast, and the "moving average" term q: presents the lags of the forecast errors. The ARIMA framework introduces an extra parameter d, and performs time series forecasting for nonstationary series. An ARIMA (k, d, q) forecasting model acts as a "signal filter" and a trend "filter" that uses d^{th} order differences to induce past observations into future forecasts.

$$\nabla^d X_t = \epsilon_t + \sum_{i=1}^{k} \alpha_i \nabla^d X_{t-i} + \sum_{i=1}^{q} \beta_i \epsilon_{t-i}; \qquad \tilde{X}_t = \nabla^d \tilde{X}_t + \sum_{i=1}^{d-1} \nabla^i X_{t-i} \quad (4)$$

We studied an application of ARIMA forecasting technique in our previous work, where soft-errors from a running scientific simulation can be identified through time-series analysis [7]. Because computational simulations such as the Brittle Fracture Model do not incur noise, term q can be set to zero $(q = 0)$. We focus on terms d and k in below sections.

Determine differencing term '*d*' We apply the Augmented Dickey-Fuller Test (ADF) [7] on our time-series. ADF's p-value was 0.689 (higher than the significant level 0.05), indicating the non-stationarity. Fig.1 shows that a 1st order of differencing (d=1) is appropriate for our time series.

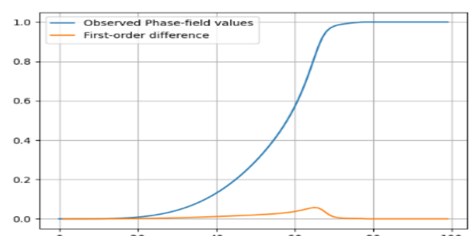


Fig. 1: Observed values per 100 timesteps (y/x-axis) with 1^{st}order differencing.

The autoregressive term '*k*' Because our forecasting approach handles data observations arriving sequentially and updates the models simultaneously, which is more natural for many time-step based applications, $k = 1$ (i.e. AR(1)) is applicable for ARIMA model. As a result, we has the ARIMA(1,1,0) model which is coined the "differenced first-order autoregressive model".

Applying ARIMA to Phase-field value prediction Because ARIMA is a linear regression-based approach, we develop an algorithm to perform multi-step out-of-sample forecast with re-estimation (i.e., the model is re-fitted each time the model makes a prediction) [8]. Accordingly, the algorithm 1 below takes a set of Phase-field values obtained from a BFM simulation with a parser grid,

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_36

builds a forecast model, performs the forecast, and reports the RMSE of the predictions. We describe how train and test sets are configured (i.e., the training portion rate R) along with the performance metrics in section 4 below.

3.2 Deep-learning based time-series modelling with LSTM

Recurrent Neural Network (RNN) uses sequential observations and learns from the earlier stages to forecast future trends. Especially, Long Short-Term Memory neural networks (LSTMs) captures and processes past and present data items in a form of a transport line. Each cell is equipped with "gates" to allow data items to be either disposed, filtered, or added to the next cells. Each sigmoid layer produces outputs between [0, 1]. LSTM network consists of:

- **Forget Gate**: outputs 0 when the data item should be completely ignored.
- **Memory Gate**: selects a new data item to be stored in the cell.
- **Output Gate**: decides what a cell will output.

Algorithm 1 : Pseudo code for multi-step forecast of Phase-field value.
Input: A_{init}: initial time-series dataset, A: the time-series dataset to be predicted, and R: training portion rate
Output: average RMSE value of the forecasted Phase-field values

```
procedure ARIMA(history:)
    model ← ARIMA(history, order=(1, 1, 0))
    model fit ← model.fit()
    return model_fit.forecast()
#Split data into train set and test set
size ← length(A); train ← A[0..size] * R; test ← A_init[length(train)..size]
observed ← train; predicts ← (); t ← length(train) + 1
while t < size then
    #build ARIMA time series model and deliver forecast
    predict.append(arima(observed))
    observed.append(test[t]); t ← t+1
end while
return sqrt(mean square error(predicts,observed[length(train)..size)])
```

With LSTM, the process of training a subsequence in a continuous, non-ending time-series might break down. Therefore, the internal state values of the LSTM network must be reset by a forget gate at an appropriate time. We enhanced the LSTM architecture [9] with a forget gate embedded to reset the internal state value for each subsequent prediction (Fig.2). Specifically, the internal connections of the LSTM consist of 4 cells in each layer. The forget gate f_t acts as the filter to remove unnecessary information. When $f_t = 0$, the information is discarded, and when $f_t = 1$ the necessary information is retained. Also, we assume that the data generated by the BFM process bias-free. Thus, the performance of LSTM cells does not depend on the bias term.

Applying LSTM for Phase-field variable prediction The overall process to apply LSTM for Phase-field value prediction is different from the ARIMA algorithm, where we use a set of Phase-field values obtained from a BFM simulation with a parser grid for training the LSTM model. Similar to the ARIMA algorithm, we output the Root Mean Square Error (RMSE) of the predictions.

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_36

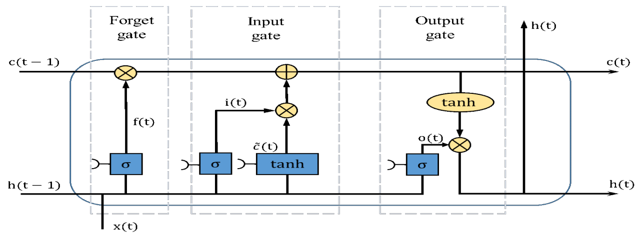
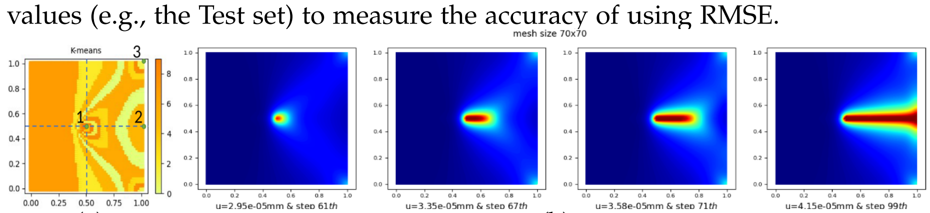


Fig. 2: LSTM cell with a forget gate.

4 Case Study - Single edge notched tensile

We consider a two-dimensional plane strain plate with size 1x1mm with a crack described as a line at the left of the plate and positioned as (Fig. 3 (a)). The material parameters are specified as: the Young's modulus $E = 210kN/mm^2$, the Poisson's ratio $\nu = 0.3$, the critical energy release rate is $\mathcal{G}_c = 5N/mm$ and the length scale parameter $l_c = 2mm$. The plate is fixed at the bottom edge and the load-displacement u at the top edge consists of 100 iterations with the increment of displacement $\Delta u = 5e-7mm$. In this example, the specimen with four mesh sizes and gradually increases the number of elements as 40x40, 50x50, 60x60, and 70x70. In this fracture model, the Phase-field variable is significant to describe the formation of cracks. With mesh size 70x70, the values of the phase-field variable in the vicinity of the crack will be affected by l_c (Fig.3 (b)), which determines the width and the diffusion of cracks. In this case study we keep $l_c = 8h$, with h being the size of each rectangular element. After the phase-field variable of the sample changes, the displacement–reaction force relations of mesh sizes are similar in the trend of behaviors.

The data of Phase-field variables at various mesh sizes are collected. We vary the size of the Train set from 30%, 40%, and 50% of the data. We simulate 100 timesteps in our fracture simulation as 30% means 30 historical time-steps. We observe how increasing the amount of training data affects the performance of different time-series forecasting methods. We use the remaining data values (e.g., the Test set) to measure the accuracy of using RMSE.



(a) (b)

Fig. 3: (a) Observed points in the plate, (b) The propagation of the crack.

5 Results

In this experiment, some critical points are observed by using K-means clustering to validate the utility and advantages of LSTM and ARIMA in the simulation. However, the point in center of the plate is picked because it is the initial position of crack propagation where the phase-field value varies from 0 to 1 over the period of simulation. Once a time-series forecasting model is trained (using either 30% to 50% of the recorded Phase-field values), we keep predicting Phase-field values across 100 time-steps using either the LSTM model or

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_36

the ARIMA model. All prediction values are stored and later used to compute the average RMSE value against the recorded Phase-field values (observations).

5.1 Prediction accuracy rate

The time-series data of the phase-field variable at three point (1),(2),(3) at Fig.3 (a) are shown with respect to Fig.4 (a),(b),(c) and Fig.4 (d),(e),(f). Overall, the results indicate that the statistical method (captured by red-dashed lines) outperforms the deep-neural network approach (captured by blue-dashed line) in all configurations (mesh size combining with different amounts of training data). ARIMA's core principle is to describe the autocorrelation of a time-series, thus it performs better given sudden changes in the changing rate of a time series (for example, around time-step 65). Fig.4 depicts the ARIMA one-step-ahead efficient forecast for this sort of behavior in our Phase-field variable. Finally, the figures as well as the RMSE results, show that ARIMA forecasting performs consistently, especially when training data is limited.

Deep learning methods such as LSTM requires more training data. However, our LSTM model can over-fit when the extrapolating time series presenting a trend. Consider the mesh size 40x40 (Fig.4 (a),(b),(c)). Our LSTM model performs quite well as the LSTM curves track closely comparing to the observation curve. However, as we increase the mesh size, our LSTM model fails to predict the plateau in Phase-field values from time-steps 65 and beyond. As Fig.4 (d),(e),(f) show this over-fitting behavior as predictions shoot above or below the expected Phase-field values.

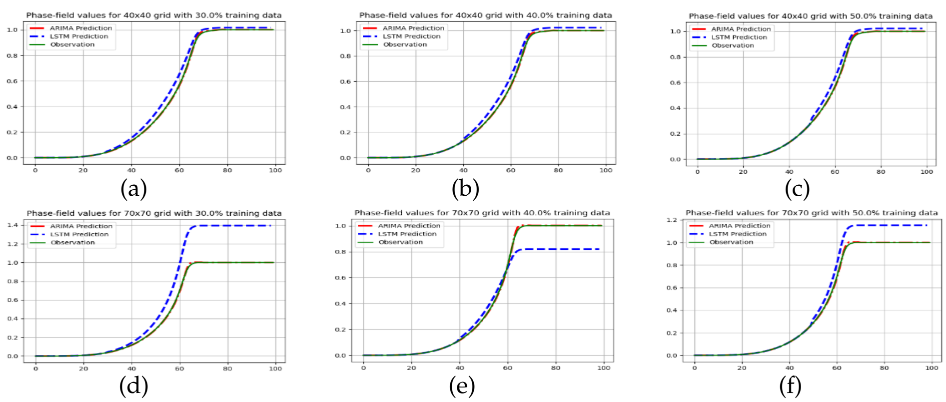


Fig. 4: Forecasting Phase-field values (y-axis) at different steps (x-axis) for mesh size 40x40 at (a),(b),(c) and mesh size 70x70 at (d),(e),(f).

5.2 Runtime performance of the prediction models

We observed that the computational costs for both machine learning models depend on two factors: the size of the mesh and the Phase-field value data used for training. Fig. 5 shows that increasing the amount of training data (30% to 50%) increases the runtime for both models. ARIMA's runtime increases around 2000 milliseconds between 30% and 50%, while it was only an increase of 800 milliseconds for the LSTM model. More importantly, with LSTM, as we increase the mesh size, the training time remains relatively constant. ARIMA model, on the other hand, shows a decrease in training time as the mesh size

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_36

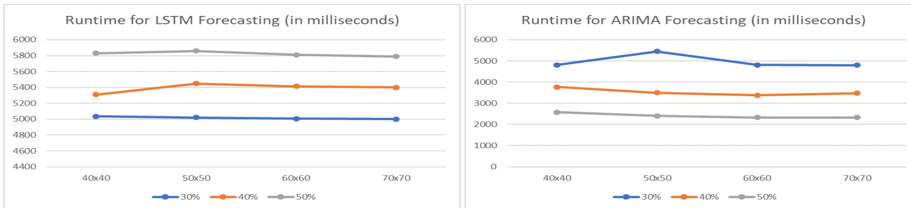


Fig. 5: Runtime (in milliseconds) for time-series forecasting models.

increases from 40x40 to 50x50 and 60x60. Overall, ARIMA's runtime is still less than LSTM's runtime regardless of the size of data. That shows that statistical approaches for time-series forecasting are still more favorable at large scales.

6 Conclusion and Future Work

The use of the phase-field variable in the illustration of crack propagation is one of the state-of-art methods. The properties of a brittle fracture model could be considered as a time-series for an online machine learning analysis and prediction. In this paper, we described how forecasting approaches were used to track the changes of the Phase-field value evaluated the performance of techniques. Our results show ARIMA delivers higher precision and accuracy and is cheaper to train and faster to fit in predicting Phase-field values, compared to LSTM. However, this work is limited to predicting Phase-field values from running brittle fracture simulations with varying mesh sizes. For future work, we also apply machine learning techniques to identify regions with the concentration of high values of the Phase-field variable.

References

1. A. A. Griffith.: "VI. The phenomena of rupture and flow in solids," Philosophical transactions of the royal society of London. Series A, containing papers of a mathematical or physical character, vol. 221, no. 582-593, pp. 163-198, (1921).
2. N. Moës, J. Dolbow, and T. Belytschko, "A finite element method for crack growth without remeshing," International journal for numerical methods in engineering, vol. 46, no. 1, pp. 131-150, (1999).
3. E. S. G. Jr, "Exponential smoothing: the state of the art," Journal of Forecasting, vol. 4, pp. 1-28, (1985).
4. Y. Yu, X. Si, C. Hu, and J. Zhang, "A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures," Neural computation, vol. 31, pp. 1235-1270, (2019).
5. G. Molnár and A. Gravouil, "2D and 3D Abaqus implementation of a robust staggered phase-field solution for modeling brittle fracture," Finite Elements in Analysis and Design, vol. 130, pp. 27-38, (2017).
6. J. D. Hamilton, Time Series Analysis. Princeton: Princeton University Press, (1994).
7. M. N. Dinh, C. T. Vo, and D. Abramson, "Tracking scientific simulation using online time-series modelling " in 20th IEEE/ACM International Symposium on Cluster, Cloud and Internet Computing, Melbourne, Australia, (2020).
8. R. J. Hyndman and G. Athanasopoulos, Forecasting: Principles and Practice, (2018).
9. C. T. Vo, M. N. Dinh, and E. Dimla, "Predicting Phase-field Behavior of Brittle Fracture Model based on LSTM Time Series Forecasting Model," in IEEE International Conference on Research, Innovation and Vision for the Future, Ho Chi Minh City, Vietnam, (2020).

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_36