16th World Congress in Computational Mechanics (WCCM)
21-26 July 2024, Vancouver, Canada
A. Korobenko, M. Laforest, S. Prudhomme, R. Vaziri (Eds)

# PREDICTION OF PLURAL CRACK PROPAGATION USING DISCOVERED PDE

**G. MURAOKA[1] AND Y. WADA[2]**

[1] Graduate School of Science and Engineering, Kindai University
3-4-1 Kowakae, Higashiosaka city, Osaka 577-8502, Japan
genki.muraoka.0350@gmail.com

[2] Department of Mechanical Engineering, Kindai University
3-4-1 Kowakae, Higashiosaka city, Osaka 577-8502, Japan
wada@mech.kindai.ac.jp

**Keywords:** Fatigue crack growth, S-FEM, Equation discovery, Physics-informed neural networks.

**Abstract.** This study presents a prediction of plural crack propagation using the discovered partial differential equations. 80% of structures fracture due to fatigue failure. Therefore, the evaluation of fatigue cracks is essential. Numerical analysis is costly, and machine-learning surrogate models have been proposed. Hence, the crack propagation path and remaining life are predicted using machine learning. A dataset is obtained from the results of a crack propagation analysis using a s-version FEM combined with an automatic mesh generation technique. The input parameters are the coordinates of the four crack tips, and the output parameters are the crack propagation vector and the number of cycles of 0.25 mm. Also, physics-informed neural networks (PINNs) have been widely studied in recent years. Thus, we took inspiration from PINNs and added a regularization term of PDE discovered by AI Feynman to the loss. As a result, the loss of a validation dataset for training constrained by PDE was reduced by about 77% compared to the unconstrained loss. The error in crack length decreased from -0.50% to 0.17%.

## 1 INTRODUCTION

Fatigue failure is the cause of more than 80% of structural failures. An evaluation of the remaining life after detecting fatigue cracks is necessary. The evaluation can be conducted using approximate formulae and the reference stress method [1, 2]. Also, fatigue crack growth is simulated by numerical computations that consider governing laws and fracture mechanics criteria: stress intensity factor, Paris' law, and criterion of crack propagation direction. Numerical computations are conducted using various methods, such as the s-version FEM (s-FEM) combined with automatic mesh generation techniques [3-10]. By setting up the analysis model separately for the global mesh (the entire analysis target) and the local mesh, which represents the local area, including cracks, s-FEM reduces the computational processing required for crack propagation. However, even with the finite element method using s-FEM, it takes much time to compute a single crack propagation case. In contrast, machine learning can predict interpolated crack propagation path and its rate faster than the finite element analysis. Therefore, machine learning can efficiently evaluate a structure's integrity on a site.

The first report of a previous study demonstrated that in a single crack with an angle, the application of the data augmentation technique as regularization reduced the error [11]. The second report improved the prediction accuracy by applying crack coalescence conditions in plural crack with different levels [12]. Furthermore, in a follow-up report, it was possible to predict the crack with different levels in the plural crack with a prediction error of less than 0.07% by considering the physical quantities [13]. In addition to our study, many other studies have been carried out to combine crack propagation and machine learning, such as using machine learning to investigate the relationship between crack propagation rate ($\mathrm{d}a/\mathrm{d}N$) and range of stress intensity factor $\Delta K$ [14].

Recently, physics-informed neural networks (PINNs), a framework that adds physical information to neural networks, have been studied widely [15, 16]. PINNs can predict with high accuracy even on small dataset, preventing neural networks from being black boxes to some extent. We were inspired by PINNs and defined by the weighted sum of the prediction loss and the PDE loss as a regularization. This regularization requires an equation for fatigue crack propagation. An ODE called the Paris' law can be applied to fatigue crack propagation:

$$\frac{\mathrm{d}a}{\mathrm{d}N} = C(\Delta K)^m \tag{1}$$

where $\Delta K$ is a range of stress intensity factor, $C$ and $m$ are material properties; however, the stress intensity factor is usually obtained by numerical analysis. It is, therefore, challenging to measure the stress intensity factor after detecting a crack in a real problem. As ODE and PDE composed of parameters that are easy to measure are preferred, PDE was discovered by AI-Feynman, a symbolic regression algorithm [17, 18]. AI-Feynman runs symbolic regression given data on output parameters concerning input parameters that are partially differentiated using PyTorch's automatic differentiation package. Compared to the results without regularization, we would like to show that this definition of loss using this discovered PDE reduces the loss in the validation data set and leads to more accurate predictions.

## 2 FATIGUE CRACK PROPAGATION ANALYSIS BY S-FEM AND GENERATION OF TRAINING DATASET

Static elastic crack propagation analysis was performed using s-FEM. Figure 1 shows the specifications for crack propagation analysis.

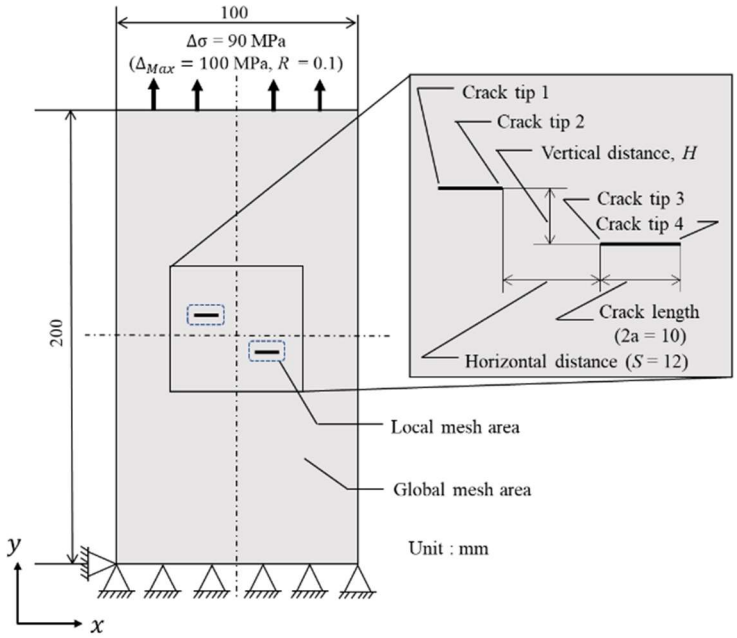


**Figure 1**: Specifications of crack propagation analysis using s-FEM.

Two different level parallel cracks (2a=10mm) are placed near the center of a 100mm x 200mm model. The horizontal distance between the two cracks is S=12mm. The vertical distance between cracks is variable $H$ mm. The model is exposed to cyclic stress in the vertical direction with a stress ratio of 0.1 and a maximum tensile load of 100 MPa. Each crack tip is called Crack tip 1 ~ Crack tip 4 from left to right. Since the amount of crack growth in one cycle is extremely small, the crack propagation analysis is performed by calculating how many cycles are required to reach a minimum mesh size of 0.25 mm at a crack tip. Twenty-one cases are simulated for 60 steps by modifying the variable $H$ from 1 mm to 36 mm according to Equation (2). One step was defined as the propagation by a minimum mesh size of 0.25 mm.

$$H_{No} = 1 + 0.25(No - 1) \qquad (1 \le No \le 6)$$
$$H_{No} = H_{No-1} + \sum_{k=6}^{No} 0.25(k - 5) \qquad (6 \le No \le 21) \tag{2}$$

The subscript $No$ of $H_{No}$ in Equation (2) represents the number of analysis cases; $H_{19}$ represents the distance between vertical cracks in analysis case 19, which is 28.25 mm from Equation (2). When the cracks are closer together, the crack interaction is so strong that a slight change in the vertical crack distance results in very different crack propagation behavior. Therefore, there is more data when the cracks are close together than when the cracks are far away.

Numerical analysis cannot represent crack coalescence. After the crack intersects, a number of cycles becomes unstable. Training dataset, including data on a number of cycles after the crack intersection, have been proven to have a negative influence on prediction accuracy [12]. Equations (3) or (4) are called crack coalescence conditions, and if either is satisfied, the two cracks are considered coalescing, and the subsequent dataset is deleted. Equations (3) and (4) are defined by the Nuclear Equipment Maintenance Standard by the Japan Society of Mechanical Engineers.

$$if\ S \leq 5\ mm, then\ H \leq 10\ mm \tag{3}$$

$$if\ S > 5\ mm, then\ H \leq 2S \tag{4}$$

Figure 2 shows the training and validation dataset and the number of effective steps after applying the crack coalescence condition. ● is the training dataset. ▲is the validation dataset.

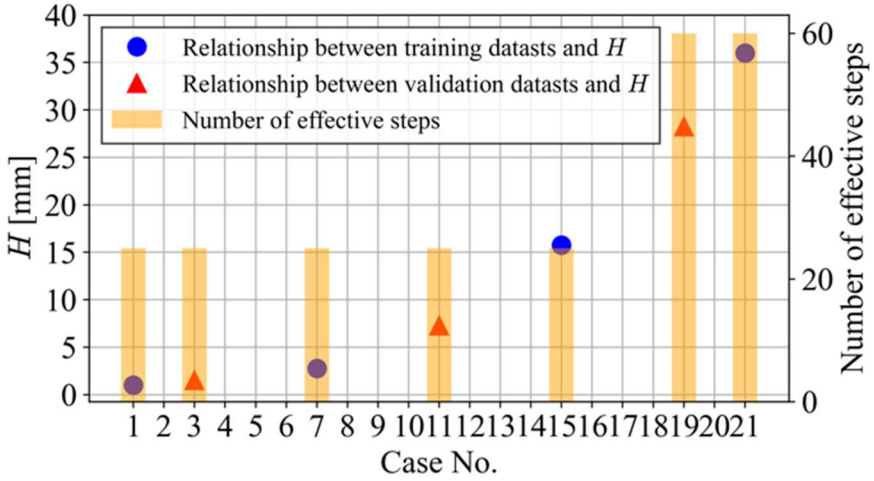


**Figure 2:** Configuration of training and validation dataset and number of effective steps.

For example, the analysis results for Case 19 are shown in Figure 3. Each crack tip propagated vertically in the direction of loading, and the number of cycles required for crack propagation decreased with increasing steps. These trends are observed in all other cases.

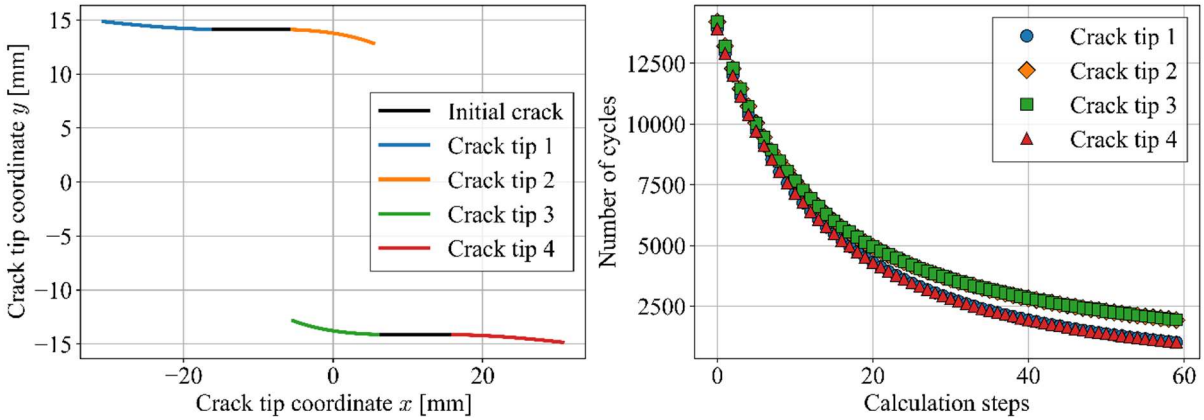


**Figure 3:** Analysis results of crack propagation path and number of crack propagation cycles in Case 19.

## 3 CRACK PROPAGATION PREDICTION METHOD

Input and output parameters are shown in Table 1.

**Table 1**: Input and output parameters.

| | |
|---|---|
| Input | Coordinate $x_i$, $y_i$ |
| Output | Crack propagation vector $t_{x_i}$, $t_{y_i}$ |
| | Crack propagation cycle $\log_{10} N_i$ |

The input parameters are the $x_i$ and $y_i$ coordinates of Crack tip $i$. The output parameters are the direction vectors $t_{x_i}$ and $t_{y_i}$ of the crack propagation of Crack tip $i$ and the number of cycles $\log_{10} N_i$ required for Crack tip $i$ to propagate to the next step. The log of $N_i$ to the base ten allows for a more linear transition in the number of cycles. Each parameter of the input and output is normalized. In predicting crack propagation, instead of predicting the crack path itself, the crack tip coordinates are predicted for each sequence of analysis steps using equations (5) and (6).

$$x_{i,n+1} = x_{i,n} + 0.25 \times \frac{N_{\min(n)}}{N_{i,n}} \times \frac{t_{x_{i,n}}}{\sqrt{\left(t_{x_{i,n}}\right)^2 + \left(t_{y_{i,n}}\right)^2}} \tag{5}$$

$$y_{i,n+1} = y_{i,n} + 0.25 \times \frac{N_{\min(n)}}{N_{i,n}} \times \frac{t_{y_{i,n}}}{\sqrt{\left(t_{x_{i,n}}\right)^2 + \left(t_{y_{i,n}}\right)^2}} \tag{6}$$

$N_{\min(n)}$ is the minimum number of cycles among the $n$-th step Crack tip $i$. The growth of each crack tip is obtained by multiplying the minimum mesh size of 0.25 mm by the number of cycles per step $N_{i,n}$ divided by the minimum number of cycles $N_{\min(n)}$. Each crack tip direction is calculated using the crack growth direction vectors $t_{x_{i,n}}$ and $t_{y_{i,n}}$. Firstly, the coordinates of each crack tip are given to the predictor. Secondly, the propagation rate and direction are predicted by machine learning. Finally, the next crack coordinate is calculated using the above equations. A series of these procedures are iterated to predict the crack propagation.

## 4 PREDICTION RESULTS WITHOUT ANY REGULARIZATION

Table 2 shows the configuration of the neural network.

**Table 2**: Configuration of neural network.

| Layer | Process | Input size | Output size | Layer | Process | Input size | Output size |
|---|---|---|---|---|---|---|---|
| 1 | Input | 8 | 8 | 10 | Liner | 10 | 40 |
| 2 | Liner | 8 | 200 | 11 | Activation (SiLU) | 40 | 40 |
| 3 | Activation (SiLU) | 200 | 200 | 12 | Liner | 40 | 100 |
| 4 | Liner | 200 | 100 | 13 | Activation (SiLU) | 100 | 100 |
| 5 | Activation (SiLU) | 100 | 100 | 14 | Liner | 100 | 200 |
| 6 | Liner | 100 | 40 | 15 | Activation (SiLU) | 200 | 200 |
| 7 | Activation (SiLU) | 40 | 40 | 16 | Liner | 200 | 12 |
| 8 | Liner | 40 | 10 | 17 | Output | 12 | 12 |
| 9 | Activation (SiLU) | 10 | 10 | | | | |

The loss function is given by

$$Loss = Loss_{NN} = MSE_{NN} \tag{7}$$

where at each Crack tip $i(= 1\sim4)$

$$MSE_{NN} = MSE_{t_{x_i}} + MSE_{t_{y_i}} + MSE_{N_i} \tag{8}$$

With N' is the number of data included in the training dataset,

$$MSE_{t_{x_i}} = \frac{1}{N'}\sum_{j=0}^{N'}\left(t_{x_{i,j}}^{target} - t_{x_{i,j}}^{pred}\right)^2 \tag{9}$$

$$MSE_{t_{y_i}} = \frac{1}{N'}\sum_{j=0}^{N'}\left(t_{y_{i,j}}^{target} - t_{y_{i,j}}^{pred}\right)^2 \tag{10}$$

$$MSE_{N_i} = \frac{1}{N'}\sum_{j=0}^{N'}\left(N_{i,j}^{target} - N_{i,j}^{pred}\right)^2 \tag{11}$$

The optimizer is Adam, the batch size is 8, and the learning rate is $1.0 \times 10^{-5}$. The training dataset was trained for 200,000 epochs. Figure 4 shows the loss transition, and Figure 5 shows the predicted crack propagation path and number of cycles. Table 3 shows a summary of the total crack length error and the total number of cycles error. The total crack length in this study is the horizontal distance from Crack tip 1 to Crack tip 4.

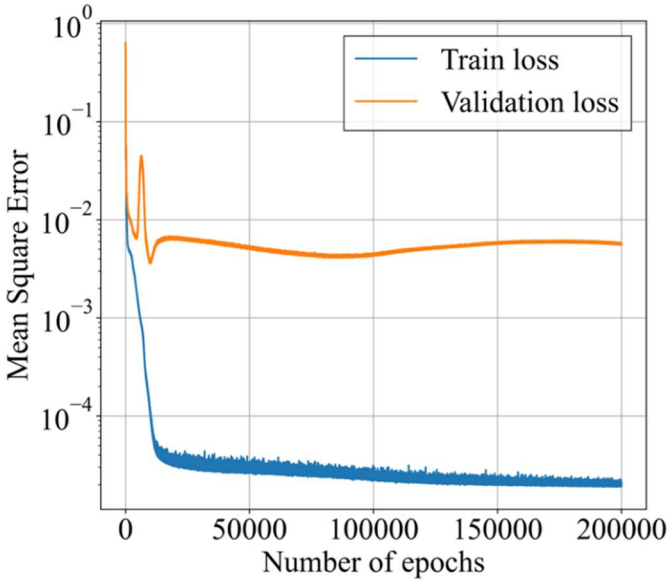


**Figure 4:** Transitions of validation and training losses.

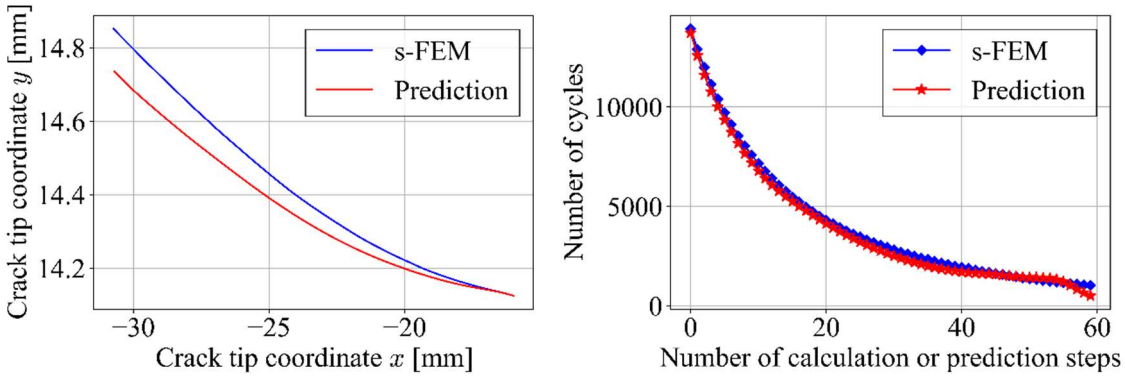


**Figure 5:** Predicted crack propagation results for Crack tip 1 in Case 19 ($H_{19} = 28.25mm$).

**Table 3**: Prediction result without regularization in Case 19.

| | Total crack length [%] | Total number of cycles [%] | | | |
|---|---|---|---|---|---|
| | | Tip 1 | Tip 2 | Tip 3 | Tip 4 |
| Without regularization | -0.50 | -5.27 | -5.05 | -5.14 | -3.86 |

Figures 5, and Table 3 show that the total number of cycle errors is high and not acceptable from an engineering point of view.

## 5 DISCOVERY OF PDE BY AI-FEYNMAN.

The crack tip propagating outwards from the model must be accurately evaluated because the length of two cracks predominantly affects the crack propagation rate. Therefore, a PDE composed of $x_1$ and $x_4$ as input parameters and $t_{x_1}$ and $t_{y_4}$ as output parameters is preferred in this study. In other words, the PDE represents the crack growth vector concerning the crack outer coordinates. For example, Figure 6 shows the transition of $t_{x_1}$ and $t_{x_4}$ concerning $x_4$, partially differentiated using PyTorch's automatic differentiation package from the results in Chapter 4. These are smoothed using a moving average after partial differentiation. It is necessary for AI-Feynman to be smoother and more symmetrical to make the equations more straightforward to discover.

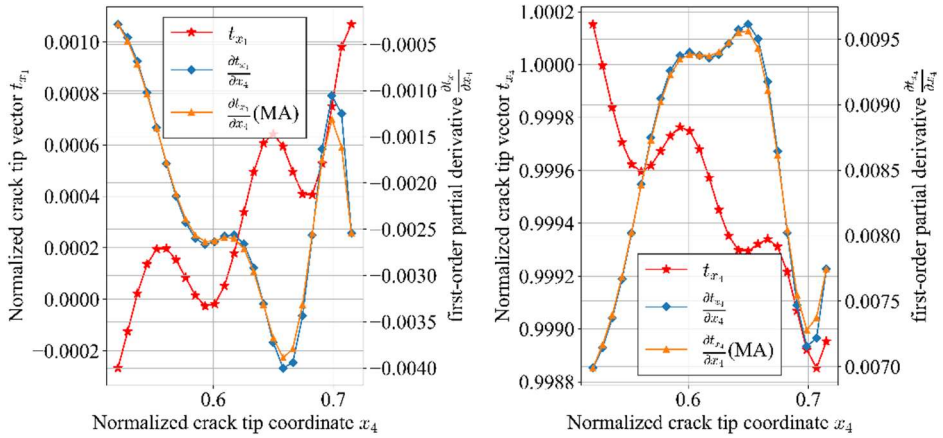


**Figure 6:** Partial differentiation of $t_{x_1}$ and $t_{x_4}$ concerning $x_4$ in Case 15.

As shown in Figure 6, the partial derivative is smooth but not symmetric. We considered that this was due to the high degrees of freedom of the neural network, so L2 regularization was applied, and the neural network was trained again ($\lambda_{L2} = 1.0 \times 10^{-6}$). Figure 7 shows the partial derivatives of $t_{x_1}$ and $t_{x_4}$ concerning $x_4$ from the training results applying L2 regularization. Training results applying L2 regularization are compared in Chapter 6.

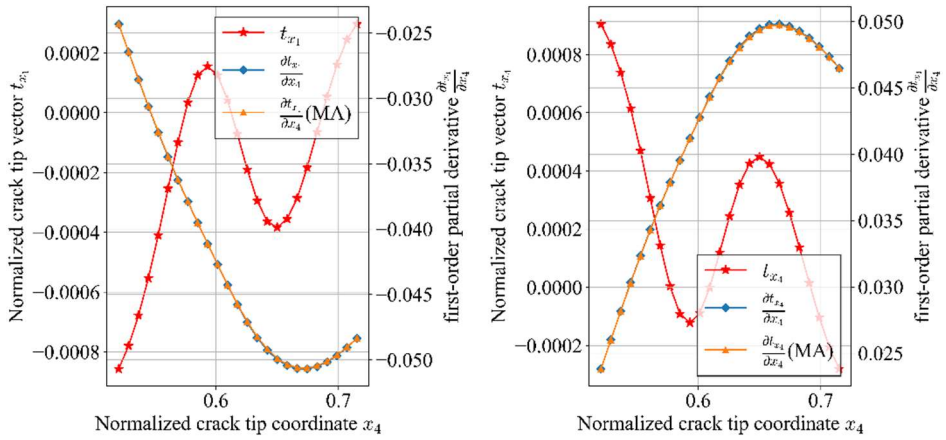


**Figure 7:** Partial differentiation of $t_{x_1}$ and $t_{x_4}$ concerning $x_4$ in Case 15 applying L2 regularization.

As shown in Figure 7, it is clear that the partial derivative is smoother and more symmetrical than without L2 regularization.

The data composed of the partial derivatives of $t_{x_1}$ and $t_{x_4}$ concerning $x_1$ and $x_4$ with L2 regularization applied are given to AI Feynman. AI Feynman can specify the symbols to be used. In this study, '+', '-', and 'exp' were specified for the interpretability and continuous and integrable nature of the equations to be discovered. The PDE discovered is shown in Equation (12).

$$\frac{\partial}{\partial x_1}\left(t_{x_1} + t_{x_4}\right) + \frac{\partial}{\partial x_4}\left(t_{x_1} + t_{x_4}\right) + 1.2934848\times10^{-6} = 0 \tag{12}$$

Considering the interpretability of Equation (12), Equation (12) is appropriate because $t_{x_1}$ and $t_{x_4}$ have opposite crack propagation directions and when added together approach zero.

## 6 PREDICTION RESULTS OF PDE REGULARIZATION

The loss function for this regularization is given by

$$Loss = Loss_{NN} + Loss_{PDE} = MSE_{NN} + \lambda_{PDE} \times MSE_{PDE} \tag{13}$$

where $MSE_{NN}$ is given by (8), and $\lambda_{PDE}$ is the weight of the $MSE_{PDE}$ and

$$MSE_{PDE} = MSE_{PDE_{t_x}} \tag{14}$$

with

$$MSE_{PDE_{t_x}} = \frac{1}{N'}\sum_{j=0}^{N'}\left(\frac{\partial}{\partial x_{1,j}}\left(t_{x_{1,j}}^{pred} + t_{x_{4,j}}^{pred}\right) + \frac{\partial}{\partial x_{4,j}}\left(t_{x_{1,j}}^{pred} + t_{x_{4,j}}^{pred}\right) + 1.2934848\times10^{-6}\right)^2 \tag{15}$$

Figure 8 shows a schematic of the PDE regularization.

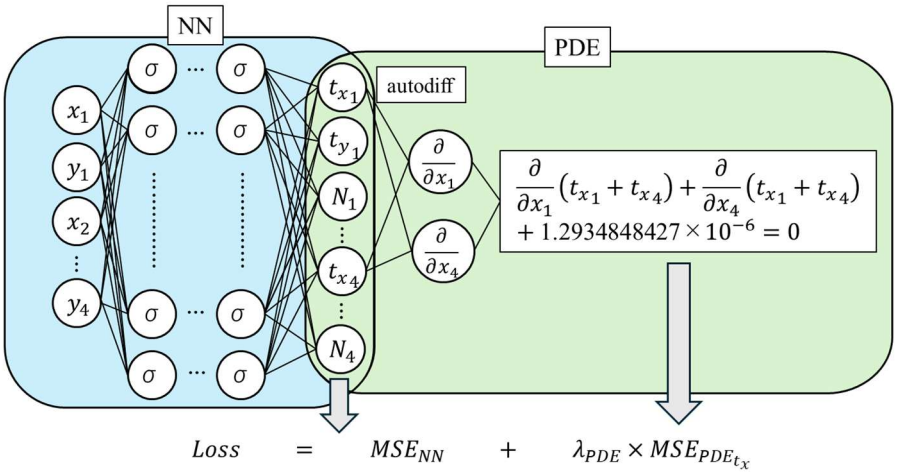


**Figure 8:** Schematic of the PDE regularization.

The only difference compared to previous training is the definition of loss. The training was conducted at $\lambda_{PDE} = 1.5 \times 10^8$. Prediction results are compared without regularization in Chapter 4 and with the application of L2 regularization in Chapter 5. A comparison of loss transitions is shown in Figure 9, and a comparison of the predicted crack propagation path and the number of cycles is shown in Figure 10. Table 4 compares the loss of the validation dataset and the errors in crack length and number of cycles. These results show that PDE regularization reduced the loss of the validation dataset by about 77% compared to without the regularization. The crack length error was also reduced to about 1/3 times. The error of the total number of cycles was reduced to about 4%, and these errors are acceptable from an engineering point of view. In addition, it shows reduced loss and improved prediction accuracy compared to L2 regularization.

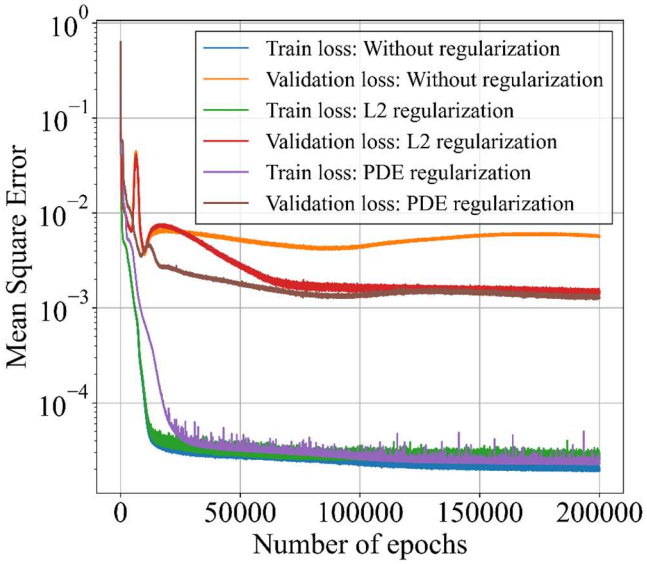


**Figure 9:** Comparison of training losses and validation losses.

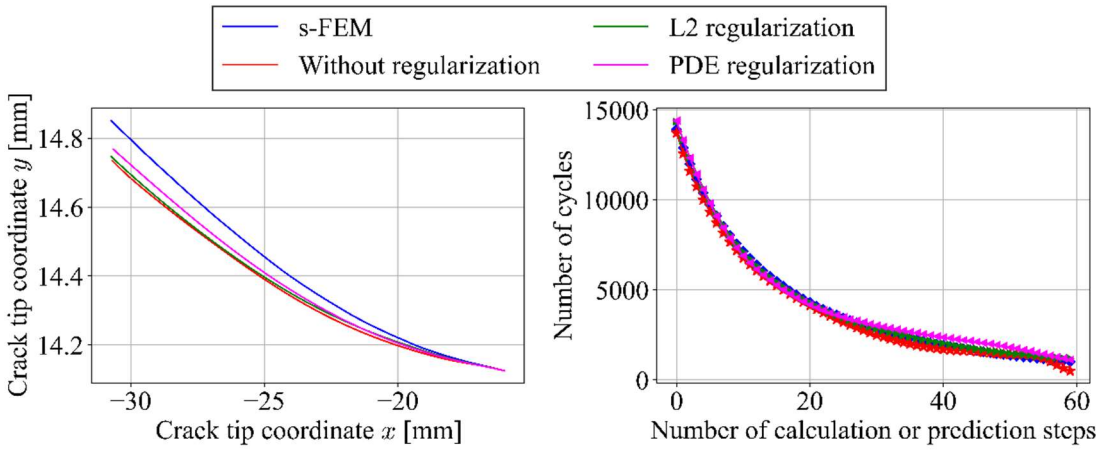


**Figure 10:** Comparison of crack propagation prediction results for crack tip1 in case 19 ($H_{19} = 28.25mm$).

**Table 4**: Summary of comparison of errors in crack length and number of cycles.

| | Validation loss | Total crack length [%] | Total number of cycles [%] | | | |
|---|---|---|---|---|---|---|
| | | | Tip 1 | Tip 2 | Tip 3 | Tip 4 |
| Without regularization | $5.64 \times 10^{-3}$ | -0.50 | -5.27 | -5.05 | -5.14 | -3.86 |
| L2 regularization | $1.42 \times 10^{-3}$ | -0.10 | 2.13 | 0.29 | -0.28 | 2.35 |
| PDE regularization | $1.27 \times 10^{-3}$ | -0.17 | 4.23 | 3.31 | 0.52 | 3.97 |

## 7 CONCLUSIONS

- The discovered PDE added to the loss function as a regularization reduced the loss in the validation dataset and improved the accuracy of the predictions. It can also reduce the validation loss compared to the L2 regularization. The result just shows the effectiveness of the simplified equation, we should study a combination of the other regularization terms to reduce total validation loss.

## REFERENCES

[1] Roche RL. Modes of failure-primary and secondary stresses. *J. Pressure Vessel Technol.* (1988) **110**(3):234‑239.

[2] Takamatsu S, Shimakawa T. Development of a simplified J-Estimation Scheme Based on the Reference Stress Method. *Journal of the Society of Materials Science*. (1994) **43**(493):1284-1994 (in Japanese).

[3] Fish J, Markolefas S. The s-version of the finite element method for multilayer laminates. *International Journal for Numerical Methods in Engineering*. (1992) **33**(5):1081-1105

[4] Fish J, Markolefas S, Guttal R, Nayak P. On adaptive multilevel superposition of finite element meshes for linear elastostatics. *Applied Numerical Mathematics*. (1994) **14**(1-3):135-164.

[5] Fish J, Guttal R. The s-version of finite element method for laminated composites. *International Journal for Numerical Methods in Engineering.* (1996) **39**(21):3641-3662.

[6] Kikuchi M, Wada Y, Takahashi M. Fatigue Crack Growth Simulation Using S-FEM. *Transactions of the Japan Society of Mechanical Engineers Series A*. (2008) **74**(742):812–

818. (in Japanese)
[7] Kikuchi M, Takahashi M, Wada Y, Li Y. Fatigue Crack Growth Simulation Using S-Version FEM (2nd Report, Study on Interaction of Two Parallel Cracks). *Transactions of the Japan Society of Mechanical Engineers Series A*. (2008) **74**(745):1243‐1248. (in Japanese)
[8] Kikuchi M, Wada Y, Utsunomiya A, Suyama H. Fatigue Crack Growth Simulation Using S-Version FEM (3rd Report, Fatigue of 3D. Surface Crack). *Transactions of the Japan Society of Mechanical Engineers Series A*. (2009);**75**(755):918‐924. (in Japanese)
[9] Kikuchi M, Maitireymu M, Sano H. Study on Fatigue Crack Growth Criterion (1st report, Paris' Law of a Surface Crack under Pure Mode I Loading). *Transactions of the Japan Society of Mechanical Engineers Series A*. (2010) **76**(764):516–522. (in Japanese)
[10] Kikuchi M, Wada Y, Suga K, Ohdama C. Effect of KIII on Fatigue Crack Growth Behavior (2nd Report, Verification of Crack Growth Criterion). *Transactions of the Japan Society of Mechanical Engineers Series A*. (2011) **77**(781):1453–1462. (in Japanese)
[11] Toyoshi T, Ozawa R, Taichi R, Wada Y. Prediction of fatigue crack growth using convolutional neural network (1st Report, Prediction for a single crack with angle). *Transactions of the JSME (in Japanese)*. (2022) **88**(915):22-00188.
[12] Ozawa R, Toyoshi T, Taichi R, Wada Y. Prediction of fatigue crack growth using convolutional neural network (2nd Report, Prediction of crack propagation on different levels). *Transactions of the JSME (in Japanese)*. (2023) **89**(924):23-00032. (in Japanese)
[13] Muraoka G, Toyoshi T, Taichi R, Wada Y. Improving the accuracy of plural crack growth prediction by machine learning considering physical quantities limited small dataset. *Transactions of the Japan Society for Computational Engineering and Science*. (2024) **2024**:20240007. (in Japanese)
[14] Mortazavi SNS, Ince A. An artificial neural network modeling approach for short and long fatigue crack propagation. *Computational Materials Science*. (2020) **185**:109962.
[15] Raissi M, Perdikaris P, Karniadakis GE. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*. (2019) **378**:686-707.
[16] Mao Z, Jagtap AD, Karniadakis GE. Physics-informed neural networks for high-speed flows. *Computer Methods in Applied Mechanics and Engineering*. (2020) **360**:112789.
[17] Udrescu SM, Tegmark M. AI Feynman: A physics-inspired method for symbolic regression. *Science Advances*. (2020) **6**(16): eaay2631.
[18] Udrescu SM, Tan A, Feng J, Neto O, Wu T, Tegmark M. AI Feynman 2.0: Pareto-optimal symbolic regression exploiting graph modularity. *Advances in Neural Information Processing Systems*. (2020) **33**:4860-4871.