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

INVESTIGATION OF CNN-BASED MULTIGRID-BIDIRECTIONAL NETWORKS

YUKIHIRO IWATA[1], YOSHIHISA INAGAKI[1] AND MIYOKO IRIKIIN [1]

[1] Panasonic Connect Co., Ltd.
ytv Kyobashi Building 2-2-33 Shiromi, Chuo-ku, Osaka City, Japan
yukihiro.iwata@jp.panasonic.com

Key words: CFD, Multi-Grid, Deep Learning, CNN

Abstract. We are developing a high-speed simulation technology for physics simulations using deep learning. This technology aims to accelerate simulation time by a factor of several hundred to a thousand, significantly enhancing product performance and quality by increasing development efficiency and optimization. Currently, we are focusing on a multi-grid convolutional neural network (CNN) based architecture designed for models incorporating a combination of coarse and dense grids, addressing the challenge of model scaling and high resolution.

Our previous work, reported by the Society for Computational Engineering and Science in June 2023, demonstrated the propagation of physical information from a coarse grid to a dense grid. Building on this foundation, we have now developed a technique that facilitates the propagation of physical information among multiple grids with varying resolutions. We applied this novel method to a basic temperature distribution prediction model for circuit boards and verified its high accuracy in predicting temperature distribution.

1 INTRODUCTION

Computational Fluid Dynamics (CFD) has become an indispensable process in the thermal design of product development before the creation of actual products. With advancements in simulators and hardware, CFD is increasingly used for evaluating large-scale models that reproduce detailed structures. However, the models we are working on often require several to several dozen hours of computation time, necessitating faster computations to respond interactively to design changes.

To address this issue, we have been working for several years on accelerating computations using deep learning, specifically Convolutional Neural Networks (CNN). This technology is now being applied to products, including implementation in our proprietary AI thermal design tool. Figure 1 shows the AI thermal design tool we have developed. This tool can calculate the temperature distribution of a circuit board in less than one second, a process that previously took more than ten minutes using traditional CFD methods. This allows for an interactive development and design process where design changes can be made while continuously checking the temperature distribution.

However, the development department requested that the tool handle larger-scale models. A CNN-based network with a fixed length of input information requires more learning time and faces insufficient GPU memory as the network scales. While the tool was initially considered

for models with up to several million elements, it was determined that handling larger scales would be challenging. In CFD, a multigrid technique is often used to reduce the number of elements and manage large-scale models. Although this method is a feature of many CFD solvers, it is difficult to apply to the CNN-based surrogate model we are developing because the input information has a fixed grid length.

To make CNNs compatible with multigrid networks, we devised a method of creating individual multigrid networks with different resolutions and fusing them to predict the temperature of the entire model. This method extends the technique we previously reported [28th Conference on Computational Engineering and Science]. In our previous work, heat propagation was limited to one direction from a coarse grid to a dense grid in a two-level multigrid. In this paper, we extend heat propagation to a three-level multigrid by considering bidirectional heat propagation.

Attempts to accelerate computations using CNN-based networks have been reported by others. Nishida et al. proposed a multi-analysis method combining CNN and domain decomposition [1]. Suzuki et al. proposed a method to accelerate Poisson solvers using CNN as a preprocessor [2]. Nakamura et al. proposed an auto-encoder type CNN for channel turbulence [3]. However, these studies differ from our approach of applying CNNs to multigrid networks with coarse and dense segmented regions.

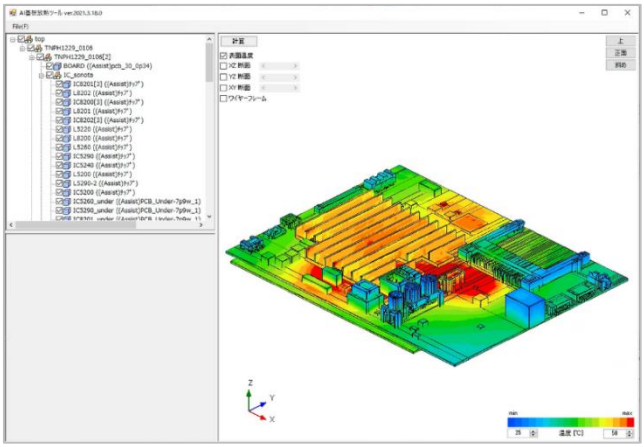


Figure1 : AI-Simulation Tool for Electric Circuit Board Heat Design.

2 CFD MODEL

Figure 2 illustrates the CFD model used to predict the temperature distribution of an electric circuit board, which serves as the computational target of this study. The model employs a three-level multi-grid structure comprising (1) parent grids, (2) child grids, and (3) grandchild (gchild) grids. The temperature distribution calculations are performed using STREAM, a commercial tool developed by Software Cradle, Inc.

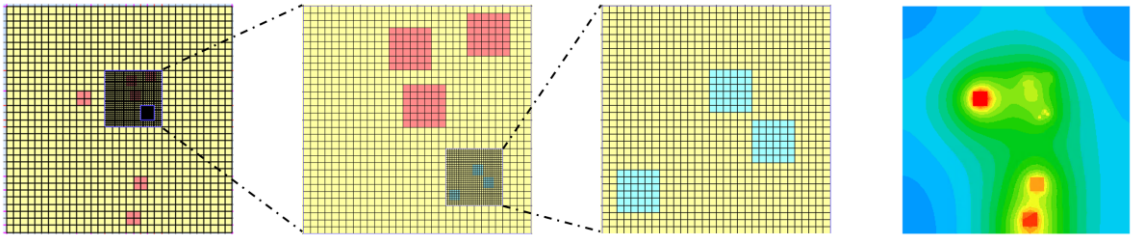


Figure2 : CFD Model

3 TRAINING DATA

Table 1 summarizes the specifications of the training data used for deep learning. Since the objective of this study is to examine the basic scheme, the model is designed with a limited set of parameters for modification. Each grid contains one multigrid, with the size of each grid being one-fourth of the parent grid size, and the position within the grid is variable. Each grid includes three integrated circuits (ICs) with fixed size and heat generation, but their positions within the grid are variable. The thermal conductivity of the substrate and ICs is 36 W/m-K, and the emissivity is set to 0. The ambient temperature is maintained at 25°C.

	part＼spec	number of parts	position	shape	size [mm]	number of grids	heat [w]
multigrid_parent	outline	fixed : 1	–	fixed : square	fixed : □160	fixed : 32 × 32	–
	IC	fixed : 3	changeable	fixed : square	fixed : □10	fixed : 2 × 2	changeable : 1
	multigrid	fixed : 1	changeable	fixed : square	fixed : □40	fixed : 8 × 8	–

	part＼spec	number of parts	position	shape	size [mm]	number of grids	heat [w]
multigrid_child	outline	fixed : 1	–	fixed : square	fixed : □40	fixed : 32 × 32	–
	IC	fixed : 3	changeable	fixed : square	fixed : □7.5	fixed : 6 × 6	changeable : 5,3,3
	multigrid	fixed : 1	changeable	fixed : square	fixed : □10	fixed : 8 × 8	–

	part＼spec	number of parts	position	shape	size [mm]	number of grids	heat [w]
multigrid_gchild	outline	fixed : 1	–	fixed : square	fixed : □10	fixed : 32 × 32	–
	IC	fixed : 3	changeable	fixed : square	fixed : □1.875	fixed : 6 × 6	changeable : 0.5
	multigrid	–	–	–	–	–	–

Table1 : Specifications of Training Data.

4 DEEP LEARNING

In this development, we have created a network based on U-Net, a type of Convolutional Neural Network (CNN). U-Net is a semantic segmentation method developed by Olaf Ronneberger et al. for biomedical applications, and it was first presented at MICCAI (Medical Image Computing and Computer-Assisted Intervention) in 2015.

Figure 3 illustrates the architecture of U-Net, which comprises an encoder and a decoder. The encoder performs multiple convolutions on the input image to extract features, while the decoder reconstructs an image of the same size as the input by performing inverse convolutions on the features extracted by the encoder. Since up-sampling to enlarge the feature map in the inverse convolution process makes it challenging to capture the positional information of objects, U-Net addresses this issue by combining the encoder's feature map with the decoder's feature map at each layer through skip connections.

The most critical aspect of U-Net is the skip connection, which combines the encoder feature

map with the decoder feature map. This operation enables highly accurate classification on a pixel-by-pixel basis. Based on this U-Net architecture, we are developing a network to predict temperature distribution using structural information (e.g., heat generation, in-plane and out-of-plane thermal conductivity, radiation, grid widths along the XYZ axes) as inputs.

Figure 4 shows the structure of the network considered in this study. The model is based on 2D Convolution, Max Pooling, and 2D Convolution Transpose layers, with skip connections at two locations. ReLU is used as the activation function. The input data to the network is standardized, and L2 regularization is applied to prevent overfitting.

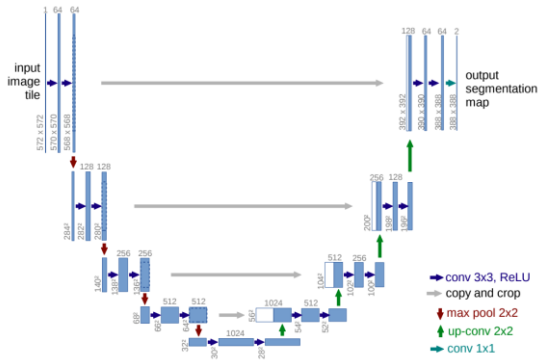


Figure3 : Outline of UNET

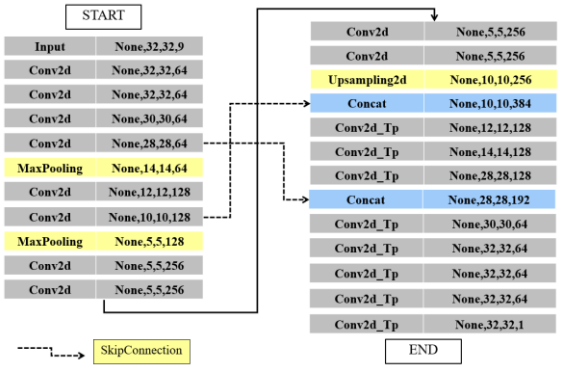


Figure4 : Network Structure of This Study

5 NETWORKS

In this study, we compare the prediction accuracy between the basic network and the series-type network, which is an improved version of the basic network. While the basic network uses the structural information of each grid as input, the series-type network incorporates intermediate temperature prediction information as additional input to enhance prediction accuracy. The details of each network are as follows.

5.1 Basic Network

Figure 5 illustrates the basic network underlying this project. The input to the network combines the structure of the parent and child grids (including heat generation, conductivity, emissivity, and mesh size), the relative position between the parent and child grids, and the relative position between the offspring grids.

Networks 3, 4, and 5 predict temperatures for the grandchild (gchild), child, and parent grids, respectively. The predicted temperatures from each network are fused to provide the overall temperature prediction.

As a preprocessing step for the parent structure, structures in the child and gchild grids other than the substrate are removed. This is because sampling the structural information in the child and gchild grids at the resolution of the parent grid is too coarse and may not accurately reproduce the structural information (especially IC size and position), which may affect learning. Similarly, the gchild grid removes structures other than the substrate.

In the temperature prediction for the parent grid, the temperatures of the child and gchild grids are not included in the loss assessment. This is because sampling the temperature information of the child and gchild grids at the resolution of the parent grid is too coarse and may not accurately reproduce the temperature information, potentially affecting learning. Additionally, there is no need to include the temperatures of the child and gchild grids in the loss assessment

because, by the time the parent grid's temperature is predicted, the temperatures of the child and gchild blocks are already determined. Similarly, the temperature of the gchild grid is not included in the loss assessment of the child grid.

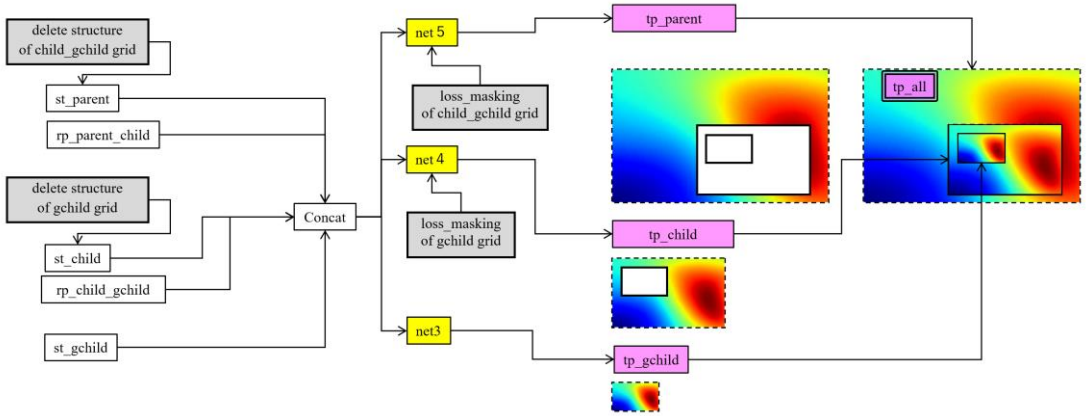
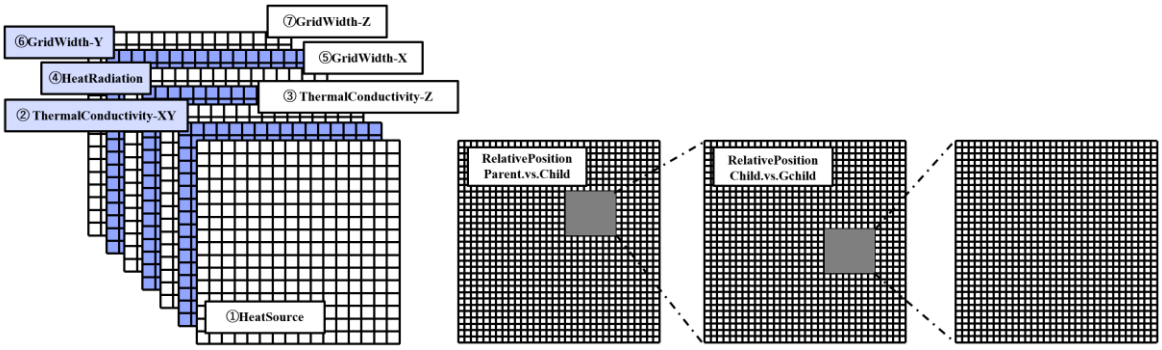


Figure5 : Deeplearning Network of Basic Model

Figure 6 illustrates the input data of the basic network.

Figure 6 (1): This panel shows the input data for the parent-child-subsidiary grid structure, which consists of seven layers related to heat generation, in-plane and out-of-plane thermal conductivity, emissivity, and grid width along the XYZ axes.

Figure 6 (2) to (4): These panels depict the parent-child relative positions and offspring relative positions. The rectangular area (gray) included in the parent grid in Figure 6 (2) corresponds to the child grid in Figure 6 (3), and the rectangular area (gray) included in the child grid in Figure 6 (3) corresponds to the subsidiary grid in Figure 6 (4).



(1)Input data (2)ParentGrid (3)ChildGrid (4)GchildGrid

Figure6 : Input data of Basic Model

5.2 Series network

Figure 7 illustrates the series-type network, an improved model of our current efforts. The primary enhancement over the basic network is the addition of Network 1 and Network 2 for predicting intermediate parent and meson temperatures. The inclusion of these intermediate temperatures aims to improve the accuracy of temperature predictions. The temperature

prediction flow is described in the following steps:

Network 1: Predicts the intermediate parent temperature from the parent structure.

Network 2: Predicts the intermediate child temperature from the child structure, using the intermediate parent temperature and the parent-child relative positions as additional inputs.

Network 3: Predicts the grandchild (gchild) temperature from the gchild structure, using the intermediate child temperatures and the relative positions of the offspring as additional inputs.

Network 4: Predicts the child temperatures from the intermediate child temperatures, using the gchild temperature and the relative positions of the offspring as additional inputs.

Network 5: Predicts the parent temperature from the intermediate parent temperature, using the child temperatures and the relative positions of the parent and child as additional inputs.

The preprocessing for the parent-child structure and the loss mask in the parent-child temperature prediction are implemented in the same manner as in the basic network.

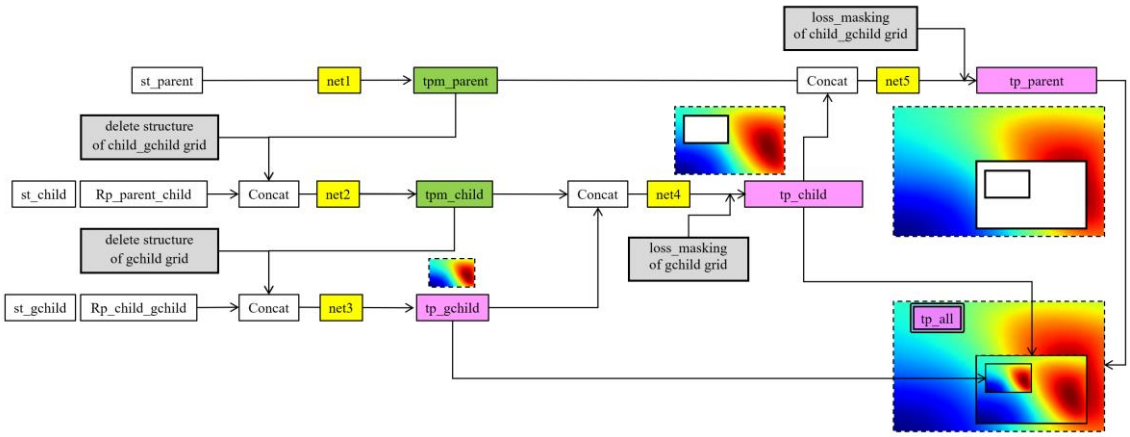


Figure7 : Deeplearning Network of Serial Model

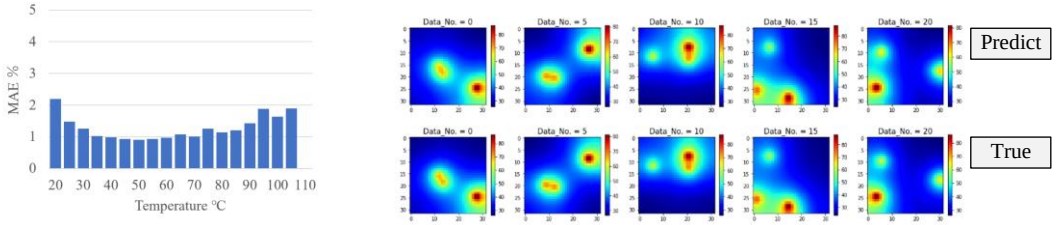
6 RESULTS AND DISCUSSION

We compare the prediction accuracy of parent, child, and grandchild (gchild) grid temperatures between the basic network and the series-type network. For the series-type network, we conduct a detailed examination of the impact of the input information on the accuracy of the temperature predictions.

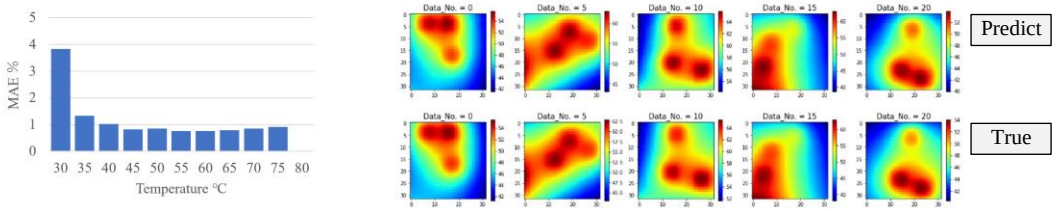
6.1 Prediction Accuracy of Series Networks

The series-type network consists of Networks 1, 2, 3, 4, and 5. Figure 8 presents the prediction results for each network. The left panel displays the prediction accuracy of the temperature distribution, where the temperature range is divided into bands at 5°C intervals, and the average prediction accuracy in each band is represented as Mean Absolute Error (MAE) in percentage. The right panel compares the predicted (upper panel) and actual (lower panel) temperature distributions.

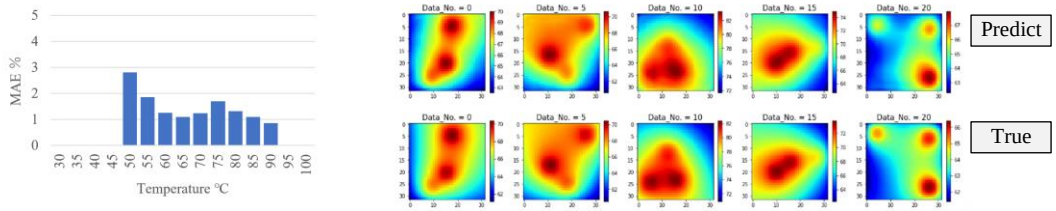
The figure demonstrates that the parent-subsidiary grid and intermediate temperatures are predicted by each network within a 5% error margin, indicating that the temperature distributions are well-reproduced. However, the prediction accuracy tends to decrease in the low and high temperature zones, which can be attributed to the lack of training data in these regions.



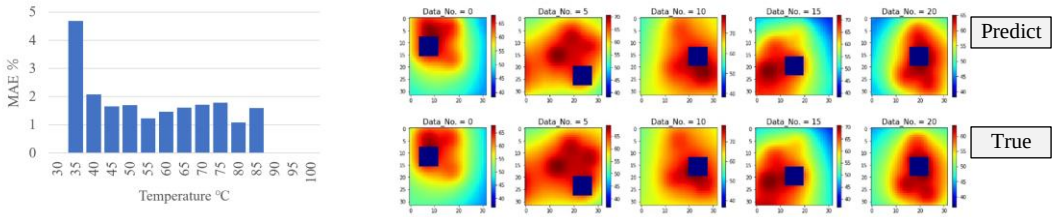
(1) Network1 : Temperature of Parent (Intermediate Predicted)



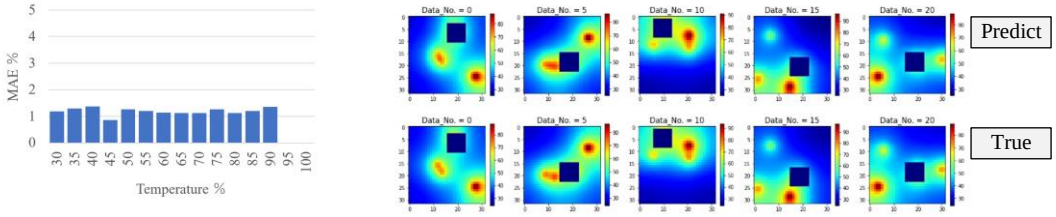
(2) Network2 : Temperature of Child (Intermediate Predicted)



(3) Network3 : Temperature of Gchild



(4) Network4 : Temperature of Child



(5) Network5 : Temperature of Parent

Figure8 : Comparison between True and Predicted Temperature by Serial Model

6.2 Input Data Validation for Series Networks

The following section describes the results of our investigation into the effect of input information on the prediction accuracy of Network 2. The fundamental approach of Network 2 is to predict child temperatures by inputting the child structure, though various combinations are possible, including intermediate predicted temperatures.

Table 2 summarizes the data combination patterns, and Figure 10 illustrates the temperature predictions for each type of data, divided into temperature bands (at intervals of 5°C), with the average prediction accuracy in each temperature band indicated by Mean Absolute Error (MAE, %).

TYPE1: Child structure plus the intermediate predicted temperature (entire child grid).
TYPE2: Child structure plus the intermediate predicted temperature (child grid area).
TYPE3: TYPE2 plus the parent-child relative position.
TYPE4: Child structure plus the parent structure and the parent-child relative position.
TYPE5: TYPE4 plus the intermediate predicted temperature (entire child grid).

TYPE3, which combines structure with the intermediate predicted temperature for the child grid region, achieves the smallest MAE. In contrast, TYPE1, where the intermediate predicted temperature for the child grid region in TYPE3 is replaced with that of the entire child grid, results in a substantially higher MAE, more than twice that of TYPE3. This increase in error can be attributed to the inclusion of information from regions that do not influence the child grid, indicating that focusing on the child grid region effectively reduces unnecessary information. TYPE4, which predicts solely based on structure without using the intermediate predicted temperature, also results in an MAE more than double that of TYPE3. TYPE5 attempts to improve upon TYPE4 by incorporating the intermediate predicted temperature for the overall region. Although a slight reduction in MAE is observed, the difference is not statistically significant.

TYPE	Child Structure	Parent Structure	Temp_Parent (ALL)	Temp_Parent (MGrid)	Position Parent.vs.Child
TYPE1	○		○		○
TYPE2	○			○	
TYPE3	○			○	○
TYPE4	○	○			○
TYPE5	○	○	○		○

Table2 : Input Data Combination

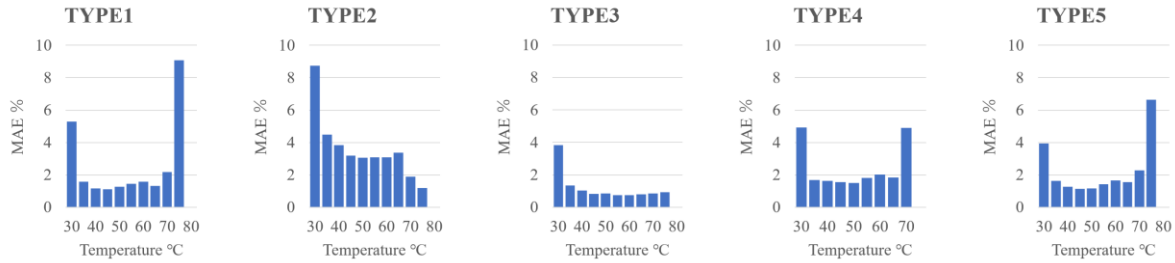


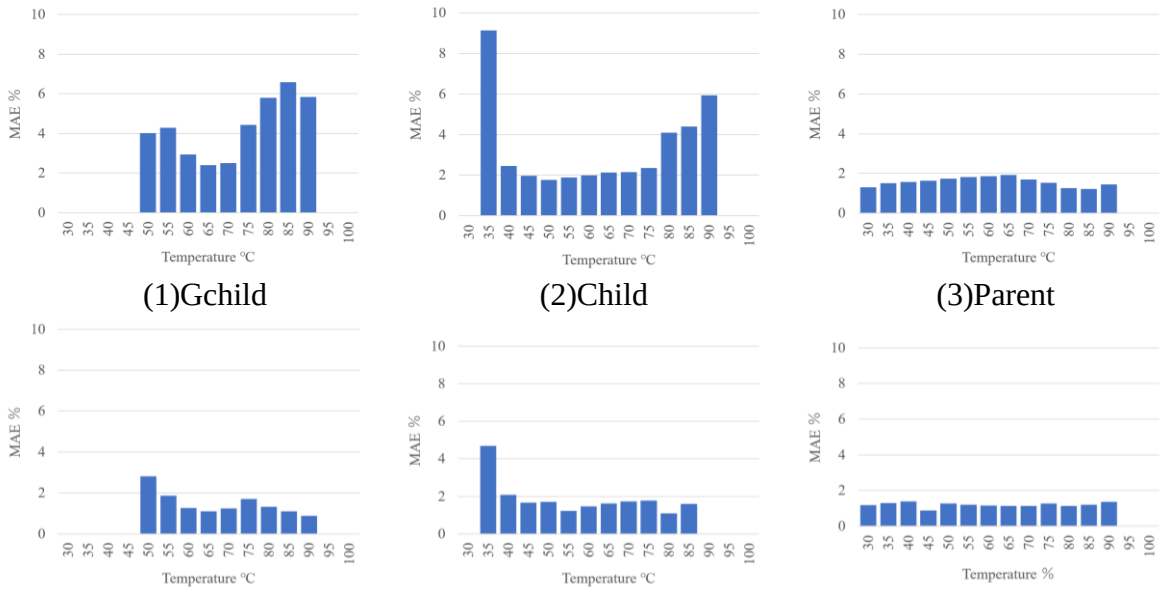
Figure9 : Effect of Input Data Types on the Accuracy of Prediction by Serial Model

6.3 Comparison of Accuracy between Basic Network and Series Network

Figure 10 compares the prediction accuracy of the basic and series networks. In the upper row, panels (1), (2), and (3) display the prediction results for the grandchild (gchild), child, and parent grids using the basic network. The average prediction accuracy for each temperature

range (in 5°C intervals) is represented as Mean Absolute Error (MAE) in percentage. Similarly, the lower row panels (4), (5), and (6) present the prediction results for the gchild, child, and parent grids using the series network.

The figure demonstrates that the series network improves the prediction accuracy by approximately a factor of 2, halving the prediction error for the gchild, child, and parent grid temperature bands.



(1)Gchild (2)Child (3)Parent

(4)Gchild (5)Child (6)Parent

Figure10 : Comparison of Basic and Serial Model

7 CONCLUSION

We have proposed a novel method for predicting the temperature of a multigrid region by creating separate networks to predict the temperature of each multigrid region and then fusing these networks to predict the temperature of the entire multigrid region. The accuracy of temperature predictions can be significantly improved by using intermediate temperatures as input information. Specifically, extracting and utilizing information from the intermediate predicted temperatures in the upper layers for the multigrid region to be predicted allows for highly accurate predictions.

In the future, we plan to extend this technology to multi-stage recursive structures, such as dense grid structures containing progressively finer grids, and to complex structural patterns, such as varying thermal conductivity distributions. We also aim to apply this method to grid regions of various sizes and shapes, including three-dimensional structures. Additionally, we intend to develop techniques for identifying effective information to enhance the efficiency of training data generation.

REFERENCES

[1]Takeru Nishida, Yoshiro Ohsuzuki, Akira Todoroki, Yoshihiro Mizutani: "Multi-scale steady-state heat transfer analysis using deep learning," Proceedings of the Conference on

Numerical Fluid Dynamics, 2018.
[2]Takako Suzuki, Yuya Omichi, et al.: "Speeding up Poisson solver in uncompressed CFD using convolutional neural networks," Proceedings of the Symposium on Computational Fluid Dynamics, B05-1, 2018.
[3]Taichi Nakamura, Koji Fukagata, et al.: "Machine learning of channel turbulence using autoencoders," Proceedings of the Symposium on Computational Fluid Dynamics, B10-1, 2019.
[4]K. Suzukami, K. Fukagata, and K. Hira: "Machine learning 3D super-resolution analysis in channel turbulence," JSME Fluid Engineering Division Newsletter "Flow", Feb. 2020, Art. 4 (2020).
[5]O. Ronneberger, P. Fischer, and T. Brox: "U-net: Convolutional networks for biomedical image segmentation," in International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, pp. 234-241, 2015.