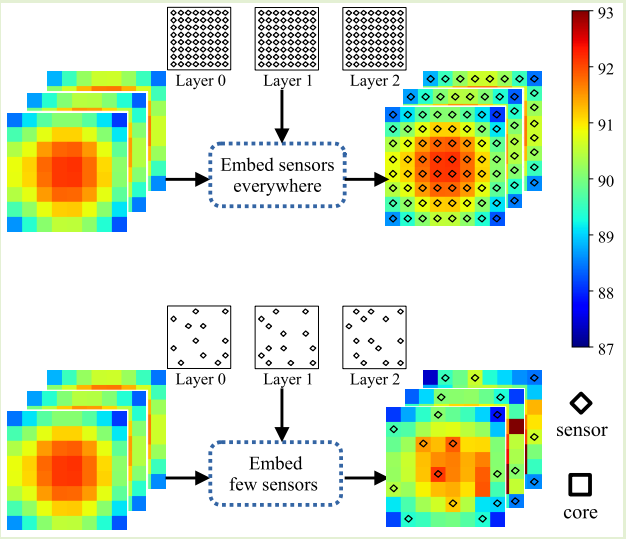
A Nearest-Neighbor-Based Thermal Sensor Allocation and Temperature Reconstruction Method for 3-D NoC-Based Multicore Systems

Menghao Guo, Tong Cheng, Xinyi Li, Li Li, *Member, IEEE*, and Yuxiang Fu, *Member, IEEE*

***Abstract*—To avoid overheating of 3-D network-on-chip (NoC)-based multicore systems, many researchers have used dynamic thermal management (DTM) techniques, which need embedded thermal sensors to provide accurate temperature information. However, only a few sensors can be embedded due to the limited hardware cost. So, it is crucial to find an appropriate way to allocate number-limited sensors at design time and reconstruct full-chip temperature accurately using the limited temperature information. However, the relationship between the non-sensor-allocated nodes and the sensor-allocated nodes modeled by the existing methods has a deviation from the actuality, which leads to an inaccurate temperature reconstruction. Another problem is that the existing methods depend highly on the training data. The estimation error can be significant when the running application's traffic characteristic differs from the one in the offline phase. This article presents a sensor allocation method based on the cores' spatial correlation, which is not dependent on the training data. Our allocation method contains two stages: 1) using our nearest-neighbor-based initialization algorithm to allocate sensors preliminarily and 2) using genetic algorithm (GA) to optimize the initial allocation. Besides, we use artificial neural network (ANN) to reconstruct the full-chip temperature. Compared with the state-of-the-art methods, our method can improve the average accuracy of the estimated temperature under different scenarios by 17.60%–88.63%. What is more, our approach has high flexibility and can adapt to different application scenarios with high accuracy with only one offline training.**



***Index Terms*—3-D network-on-chip (NoC), full-chip temperature reconstruction, soft computing with sensor data, thermal monitoring, thermal sensor allocation.**

I. Introduction

DENNARD'S law proved that if the voltage is scaled down with the size of the transistor, the electric field in the device and most device parameters can remain unchanged [1]. This law guarantees that chips' frequency can be raised as transistor size shrinks without increasing power density. Nevertheless, if the feature size is too small, Dennard's law will be invalid due to the arising tunneling current. In this case, increasing the frequency to improve the core's speed will significantly increase the power density, which means that the single core's performance reached the bottleneck, and people began to develop multicore chips to improve performance [2]. Compared to other multicore communication architectures, network-on-chip (NoC) has advantages in delay, performance, and design efficiency [3], [4]. However, the thermal problem is severe in the multicore system. Especially when the 3-D NoC-based multicore system appears, stacking dies or wafers results in a higher power density and longer heat dissipation paths [5]. Overheating leads to many problems, such as reduced performance and lifetime [6], [7].

In recent years, many dynamic thermal management (DTM) techniques have been proposed to prevent the overheating problem [8], [9], [10], [11] in NoC-based multicore systems.

Manuscript received 18 October 2022; accepted 30 October 2022. Date of publication 8 November 2022; date of current version 14 December 2022. This work was supported in part by the National Nature Science Foundation of China under Grant 62104098, in part by the Natural Science Foundation of Jiangsu Province for Youth under Grant BK20210178, in part by the Joint Funds of the National Nature Science Foundation of China under Grant U21B2032, and in part by the National Key Research and Development Program of China under Grant 2021YFB3600104. The associate editor coordinating the review of this article and approving it for publication was Prof. Chao Tan. *(Corresponding author: Yuxiang Fu.)*

The authors are with the School of Integrated Circuits and the School of Electronic Science and Engineering, Nanjing University, Qixia District, Nanjing, Jiangsu 210023, China (e-mail: yuxiangfu@nju.edu.cn).

Digital Object Identifier 10.1109/JSEN.2022.3218953

The principle of DTM is to dynamically regulate workloads of different cores based on the temperature distribution of the chip. It aims to control the temperature of the multicore chip under a threshold. The accurate temperature distribution can improve DTM's effect. The most straightforward way to obtain a full-chip temperature profile is to embed thermal sensors in each core. However, the limitation of cost and power restricts the number of thermal sensors that can be embedded [12]. So, it is essential to find a proper way to allocate the number-limited thermal sensors and reconstruct temperature distribution based on the measurements of these thermal sensors.

Many researchers have proposed many approaches to find locations to allocate the number-limited sensors [12], [13], [14], [15], [16], [17], [18]. Some methods were highly dependent on the training data. For example, the thermal sensors were embedded in the nodes with high energy in the frequency domain, where the temperature gradient is large and more temperature information is contained [14]. However, different applications may cause changes in hotspots' position and temperature gradients. Thus, this kind of approach is not flexible and can only be applied when the application is known and estimated in advance. To eliminate the dependence on the offline training data, Chen et al. [16], [17] allocated sensors using the compressive sensing theory. However, the computing delay of the reconstruction was too significant for the real-time system due to the complex algorithm of the compressive sensing theory. In terms of reconstruction, many approaches use the weighted linear combination method to reconstruct the full-chip temperature [12], [13], [14], [15]. Moreover, this linear method cannot describe the complex thermal relationship between the nodes accurately. Some approaches used more powerful models to improve the accuracy at the unacceptable cost of large hardware cost or computing delay. For example, Li et al. [18] proposed to use convolutional neural networks (CNNs) to extract high-resolution thermal maps of the multicore processor. Although this method can achieve high accuracy, it requires a large memory size to store its weights (about 150 MB), which may exceed the saved hardware cost using fewer sensors.

Based on the analysis of the previous approaches, we find that the current thermal-sensing NoC-based multicore system design has two design challenges: 1) the flexible allocation of number-limited sensors that can adapt to different applications and 2) the accuracy of full-chip temperature reconstruction that requires acceptable computing latency and hardware cost. In this article, we allocate the number-limited thermal sensors based on the spatial thermal correlation of the cores, and the relative position of the cores will not change as the chip works under different applications. Besides, we use artificial neural network (ANN) to estimate the temperature of nonsensor-allocated nodes. The experimental results show that this work improves the accuracy of the reconstructed temperature with the same number of thermal sensors compared with [12], [15], and [19]. Besides, our approach has high flexibility and can adapt to different application scenarios with high accuracy with only one offline training. Moreover, the computing latency for temperature reconstruction and hardware cost is acceptable. The contributions of this article are summarized as follows.

1) We propose the nearest-neighbor-based thermal sensor allocation method. The previous work [20] has proven that the temperature correlation of different cores in the multicore system is mainly related to their distance. Based on the spatial correlation, we propose the nearest-neighbor-based initialization algorithm to make thermal sensors surround every nonsensor-allocated core. We then use the genetic algorithm (GA) to optimize the initial sensor allocation to adjust the number of obtained sensors.
2) We propose the ANN-based full-chip temperature reconstruction method. We can get the accurate temperature of these sensor-allocated nodes from the allocated thermal sensors. Due to the temperature correlation, these nodes also contain temperature information of other nonsensor-allocated nodes around them. We can express the hidden temperature information explicitly with ANN to estimate the full-chip temperature precisely.

The organization of this article is as follows. Section II overviews some related state-of-the-art works. Sections III and IV describe the proposed thermal sensors allocation method and the ANN-based full-chip temperature reconstruction method, respectively. In Section V, we describe the hardware structure of the proposed temperature reconstruction unit. In Section VI, we evaluate the robustness, stability, and other properties of the proposed method. Besides, we compare the accuracy of reconstructed full-chip temperature and the hardware cost between our method and other related works [12], [15], [19]. Finally, Section VII summarizes the main conclusion of this work.

II. Related Works

A. Hotspot-Based Sensor Allocation and Temperature Reconstruction

Reda et al. [13] proposed a greedy algorithm to find the highest temperature nodes in the multicore system and allocate the thermal sensors on these hotspot nodes. However, when the chips are manufactured, the locations of the sensors cannot change with the running workloads. Therefore, when the chip runs under a different workload rather than the one they used to determine the sensor placement, the method cannot accurately track the temperature of hotspots due to the location mismatch between sensors and hotspot nodes. Besides, they used a weighted linear combination of hard sensors' measurements to estimate the temperature of nonsensor-allocated nodes, which are named soft sensors. The estimation method is inaccurate, especially when the hotspots change and the location mismatch occurs.

B. PCA-Based Sensor Allocation and Temperature Reconstruction

Juri et al. [15] paid more attention to the full-chip temperature instead of just the information of hotspots. They employed the principal components' analysis (PCA) and built a linear model based on PCA to allocate thermal sensors and

reconstruct the full-chip temperature. FrameSense, a greedy algorithm [21] based on the theoretical framework has also been used to determine where to embed the thermal sensors. However, this method has a strong relationship to the workloads running on the NoC-based multicore system. Besides, it also ignores the physical correlation between the nodes. Furthermore, the reconstruction model also depends on the workloads of the NoC-based multicore system because they assumed that the average chip temperature at design time is the same as the one at run time when estimating the full-chip temperature.

C. Correlation-Based Sensor Allocation and Temperature Reconstruction

Chen et al. [12] proposed to allocate the thermal sensors on the nodes with the largest sum of correlation weights with other nodes. The larger the correlation coefficients between the nonsensor-allocated nodes and the sensor-allocated nodes, the more information the sensor-allocated nodes contain from the nonsensor-assigned nodes. Besides, they proposed a linear-regression-based full-chip temperature tracking approach. They consider the relationship between the nodes to be linear, and they estimate a nonsensor-allocated node's temperature based on the sensing temperature of the highest correlation node. Nevertheless, just using linear regression cannot estimate temperature accurately when the temperature of two nodes is not in a linear relationship.

D. Co2-ANN-Based Sensor Allocation and Temperature Reconstruction

Our previous work [19] also used correlation coefficients to describe the degree of the thermal correlation between nodes. Besides, it used ANN to reconstruct full-chip temperature. ANN can estimate the temperature of the nonsensor-allocated nodes using the temperature of the sensor-allocated nodes, which have high correlations with them. We call this method the Co2-ANN-based approach, which means the combination of correlation coefficients and ANN. For ANN, the number of the input neurons equals the number of the usable thermal sensors, and the number of the output neurons equals the number of the nonsenor-allocated nodes. In this way, the size of the ANN in [19] is large, which causes high hardware cost. Besides, the correlation coefficient is a statistical indicator, and it is obtained from the temperature data in the offline phase. As shown in Fig. 1, when the training data does not contain the working scenarios that will be run on the chip, the reconstruction error will be significant due to the different correlation coefficients.

III. Nearest-Neighbor-Based Thermal Sensor Allocation Method

As introduced in Section II, the features of NoC-based multicore systems used in previous works for allocating the thermal sensors will vary with the running application, such as the locations of hotspots in [13], the correlation coefficients in [12] and [19]. In this case, the reconstruction accuracy will drop sharply when the work scenarios differ from the offline

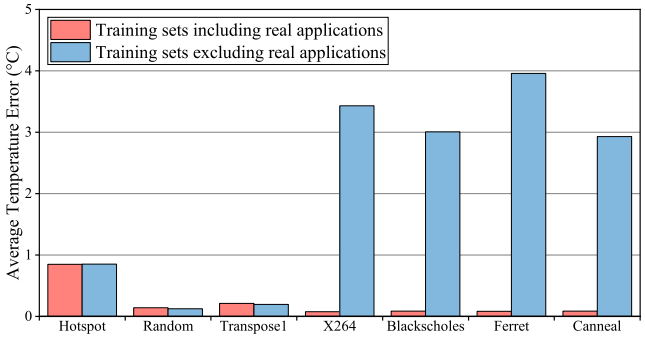


Fig. 1. Reconstruction error comparison between training sets including real applications and excluding real applications using the Co2-ANN-based method [19].

training dataset. To improve the flexibility of the temperature reconstruction, we propose to allocate the thermal sensors based on the spatial correlation, which is related to the physical structure of the NoC-based multicore system and has nothing to do with the running application. Furthermore, we propose the nearest-neighbor-based thermal sensor allocation method based on the spatial correlation.

A. Mathematical Model

To decrease the dependency on the offline dataset, we use Matérn correlation function to determine whether there is a great correlation between the nodes [20] based on their spatial distance. The Matérn correlation function is described as follows:

$$\varrho(h) = \frac{1}{2^{\theta_2-1}\Gamma(\theta_2)}\left(\frac{2h\sqrt{\theta_2}}{\theta_1}\right)^{\theta_2} \mathcal{K}_{\theta_2}\left(\frac{2h\sqrt{\theta_2}}{\theta_1}\right) \quad (1)$$

where $\mathcal{K}_\alpha()$ denotes the modified Bessel function of the second kind of order α, and $\Gamma()$ denotes the Gamma function and $h = ||l - l'||$ represents the distance between two nodes. Besides, θ_1 and θ_2 are parameters, and they do not change in one certain chip. So we can conclude that the correlation of two nodes is mainly dependent on the distance between them, and the farther their distance is, the less correlation they have. In other words, the correlation defined by (1) will not change when the chip works under different workloads. Thus, we propose a nearest-neighbor-based thermal sensor allocation method. Our method aims to allocate thermal sensors around the nodes without sensors. Because of their close distance, which means the small h, their temperature information has a high correlation. Therefore, nonsensor-allocated nodes' temperature can be estimated by the sensor-allocated nodes around them.

First, we consider that the nodes with a distance of one grid have a high correlation, such as A and B in Fig. 2. In this case, we may use six nodes' temperatures in six different directions, including east, west, north, south, up, and down to estimate the center node's temperature in 3-D Mesh NoC-based multicore systems. However, many sensors are required in this case, which leads to an enormous hardware cost for the sensors and the ANN-based temperature reconstruction unit.

To decrease the number of sensors, we take the following measures. According to the physical structure of the 3-D

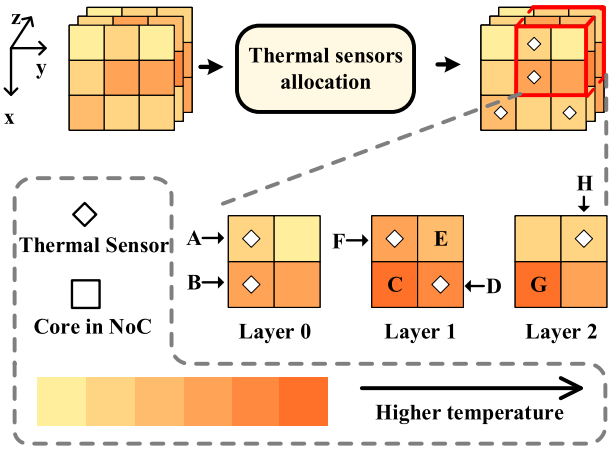


Fig. 2. Placements of the thermal sensors in a 3 × 3 × 3 NoC-based multicore system.

stacked chip, the adjacent node in the Z direction is closer than the neighbor node in the $X-Y$ direction. As shown in Fig. 2, compared with node A, node G is closer to B and has a higher correlation with it. Therefore, the nodes that are aligned in the Z direction will be highly correlated, even if they are not adjacent. For the cores in the $X-Y$ directions, we place two sensors around the nonsensor-allocated node rather than four sensors due to the symmetry of the mesh topology. In Section V, we show that using two sensors in the $X-Y$ directions has almost the same accuracy as using four sensors. Thus, the objective of our allocation can be summarized as follows.

1) Allocate thermal sensors to guarantee that every non-allocated node has one thermal sensor around it within one grid away in the X and Y directions, respectively.
2) Allocate thermal sensors to guarantee at least one thermal sensor in each Z direction.

In our model, we idealize the physical properties of sensors, such as size, sensitivity. For example, we assume that the sensor's temperature measurements are accurate, and its accuracy is not affected by the environment's temperature. These assumptions are reasonable because this article focuses on how to allocate the sensors and reconstruct the full-chip temperature distribution.

This mathematical model belongs to the graph coloring problems, and the number of sensors required to fit the above two guidelines has a minimum value. We find that using a mathematical algorithm to calculate the minimum number and find the best allocation is hard. Besides, the number of thermal sensors is a design parameter. This method cannot deal with the situation that the number of usable thermal sensors is less than the minimum. Moreover, the search space of the sensor allocation is enormous, especially when the number of nodes in a NoC-based multicore system is great. For example, when 70 thermal sensors are allocated in a 256-core NoC-based multicore system, there are about 9.3×10^{63} different allocation methods.

This article uses the GA to find the optimal sensor placement. We first propose an initialization algorithm to allocate

Algorithm 1 Nearest-Neighbor-Based Initialization Algorithm

Require: The size of the NoC-based multicore system: $l \times w \times h$
Ensure: The locations to allocate thermal sensors: $\mathcal{L}$
1: Initial $\mathcal{L} = \emptyset$;
2: Initial the available locations: $\mathcal{S} = \{1, 2, \ldots, m\}$, $m = l \times w \times h$;
3: Initial the non-sensor-allocated location set: $\mathcal{N} = \emptyset$;
4: **while** $\mathcal{S} \neq \emptyset$ **do**
5: Randomly choosing i in $\mathcal{S}$ as a non-sensor-allocated location, $\mathcal{N} = \mathcal{N} \cup i$, $\mathcal{S} = \mathcal{S} \backslash i$;
6: **for** j in $\mathcal{N}$ **do**
7: $X = findX(j)$;
8: $Y = findY(j)$;
9: $Z = findZ(j)$;
10: **if** $X == 0 or Y == 0 or Z == 0$ **then**
11: $\mathcal{L} = \mathcal{L} \cup i$, $\mathcal{N} = \mathcal{N} \backslash i$;
12: break;
13: **end if**
14: **end for**
15: **end while**
16: Return $\mathcal{L}$.

to fit the above two rules. The initialization algorithm can reduce the search space for GA. Then, we use GA to optimize the sensor allocation with the above constraints according to the number of usable sensors. This method can work when the number of usable thermal sensors is less than the minimum value required by the two rules due to our reasonable definition of fitness value for GA.

B. Nearest-Neighbor-Based Initialization Algorithm

The initialization algorithm is shown in Algorithm 1. At the beginning, we adopt a conservative strategy. To guarantee that all the nodes meet the constraints, we presume that all the nodes have embedded thermal sensors, as shown in line 2. After that, we randomly choose the location i from the candidate location set $\mathcal{S}$ and add it to the set $\mathcal{N}$, which consists of those locations that do not need to embed sensors, as shown in line 5. Then, we judge whether this change, moving i from $\mathcal{S}$ to $\mathcal{N}$, will make the allocation no longer meet the constraints. As shown in lines 7–9, for every node in $\mathcal{N}$, we check whether it has thermal sensors around it in the X, Y, and Z directions. The functions find$X()$, find$Y()$ will determine whether there are sensors in its X, Y directions within one grid distance away, and find$Z()$ will determine whether there are sensors in its Z direction. If it does not meet the constraints, we can conclude that embedding a sensor in node i is required to meet the constraints. In this case, the next step is to add j to the set $\mathcal{L}$ as shown in lines 10–11. Otherwise, there is no need to place a sensor in node j.

Although the sensor allocation obtained by the initialization algorithm can meet the constraints, the number of sensors cannot be adjusted according to the constraints of the hardware cost. Therefore, we use GA to explore the optimal sensor-allocated locations of different sensor numbers.

C. Allocation Optimization Using GA

The GA is a heuristic algorithm that aims to search for the optimal solution by simulating the process of natural evolution. In this article, we define the chromosome, gene, and fitness value as follows.

1) *Chromosome and Gene:* Each chromosome has m genes in our method, and m is the total number of cores. Each gene can be set to 0 or 1. 0 and 1 represent whether the thermal sensor needs to be embedded in this location.
2) *Fitness Value:* The definition of the fitness value f is shown in the following equation:

$$f = \begin{cases} \alpha V_n + (1-\alpha) V_c, & N \in [N_{\min}, N_{\max}] \\ 0, & \text{Otherwise.} \end{cases} \tag{2}$$

To adjust the number of sensors generated by GA, we set the search range as $[N_{\min}, N_{\max}]$. If the number of sensors N is so small or large that it exceeds the threshold $N_{\min}$ and $N_{\max}$ we set, the fitness value will be 0. If the number of sensors is within our limits, the fitness value is a weighted sum of V_n and V_c

$$V_n = \frac{N_{\max} - N}{N_{\max} - N_{\min}} \tag{3}$$

$$V_c = \sum_{i=\mathcal{N}} \frac{\text{find}X(i) + \text{find}Y(i) + \text{find}Z(i)}{3|\mathcal{N}|} \tag{4}$$

where V_n is a normalized value that evaluates the number of sensors. The smaller number of sensors leads to a higher V_n. For each sensor in the nonsensor-allocated location set $\mathcal{N}$, we count the number of sensors around it and set its average value as V_c. V_c evaluates how much the distribution of sensors meets the constraints.

Designers can set the values of all parameters, $N_{\min}$, $N_{\max}$, and α according to the number of their usable sensors. By changing the parameters in (2), our allocation method can work even though the number of usable sensors is smaller than the minimum value required by the two rules in Section III-A. When allocating the usable sensors, we can set the $N_{\min}$ in (2) to equal the number we expect and set $\alpha \geq 0.5$. In this case, our two restrictions proposed above will be relaxed, and the number of usable sensors of the best allocation will meet our expectations. For example, the sensors nearest to some nonsensor-allocated nodes are outside one grid in the X or Y direction when the number of the usable sensors is smaller than the minimum value. Although it will decrease the spatial correlation between the nodes and reduce the reconstruction accuracy, we think it is a normal tradeoff between cost and performance in the IC design.

IV. Artificial-Neural-Network-Based Full-Chip Temperature Reconstruction Technique

As mentioned before, many approaches use linear methods to reconstruct the full-chip temperature [12], [13], [15]. These methods may cause significant reconstruction errors as the correlations of the nodes are not in a simple linear relationship. In this article, we use ANN to estimate the temperature of the nonsensor-allocated nodes. ANN can learn the nonlinear relationship between the nodes more precisely and have high accuracy. Besides, we build a lookup table to describe which sensing temperatures should be used as the inputs of the ANN when estimating a nonsensor-allocated node's temperature.

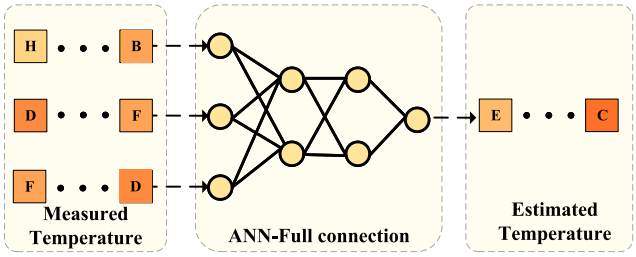


Fig. 3. Structure of ANN to reconstruct the full-chip temperature for the 3-D NoC-based multicore system in Fig. 2.

A. Structure of the ANN

After determining the locations of the embedded thermal sensors, we construct an ANN to reconstruct the full-chip temperature during runtime according to the correlation between the nodes and their neighbors. ANN can learn the correlation relationship and estimate the temperature of the nonsensor-allocated nodes based on their neighbors' sensing temperature.

The structure of ANN we use is shown in Fig. 3. It has one input layer, two hidden layers, and one output layer, and they are full connection layers. Besides, we select the rectified linear unit (ReLU) function as the activation function. The number of neurons in the four layers is 3, 2, 2, and 1, respectively. The three-input vector of the ANN is constituted of the temperatures measured by a neighbor sensor in the X direction, a neighbor sensor in the Y direction, and a Z direction sensor. The output is the node's estimated temperature. Besides, for the training, we use the

$$\text{cost} = (\text{Output} - \text{Actual})^2 \tag{5}$$

as the cost function, where Output is the output of ANN, and it is the estimated temperature. Actual is the actual temperature of the node.

B. Lookup Table Building

When estimating the temperature of a nonsensor-allocated node, its three neighbors' sensing temperatures should constitute a three-input vector of the ANN. However, when only a few sensors are available, some nonsensor-allocated nodes may not have neighbor sensors, and the ANN does not have enough input data in this case. Therefore, we find an appropriate way to determine which three measurements can be used to estimate one specific nonsensor-allocated node's temperature and use a lookup table to store the corresponding neighbor nodes whose temperatures are used as ANN inputs. The lookup table consists of four columns, which respectively represent the indexes of the nodes to be estimated and the indexes of the sensor-allocated nodes in the X–Z directions of the nodes to be estimated. The scheme to build the lookup table using $\mathcal{L}$ and $\mathcal{N}$ contains the following two steps.

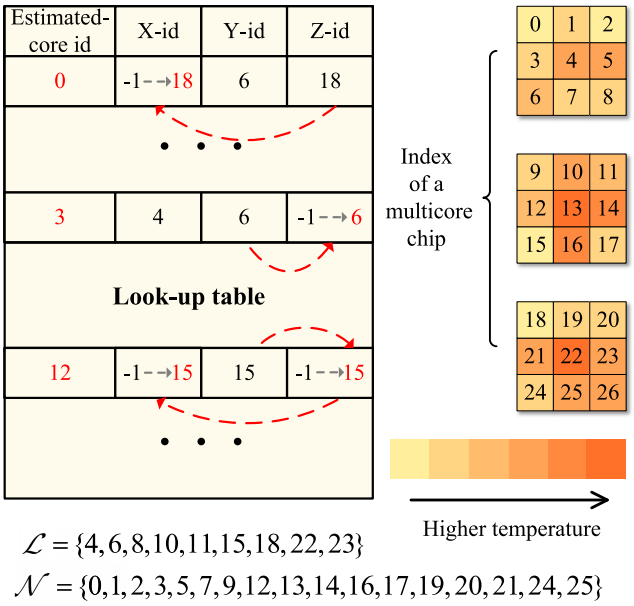


Fig. 4. Simple example of building a lookup table for a 3 × 3 × 3 NoC-based multicore system after allocating the thermal sensors.

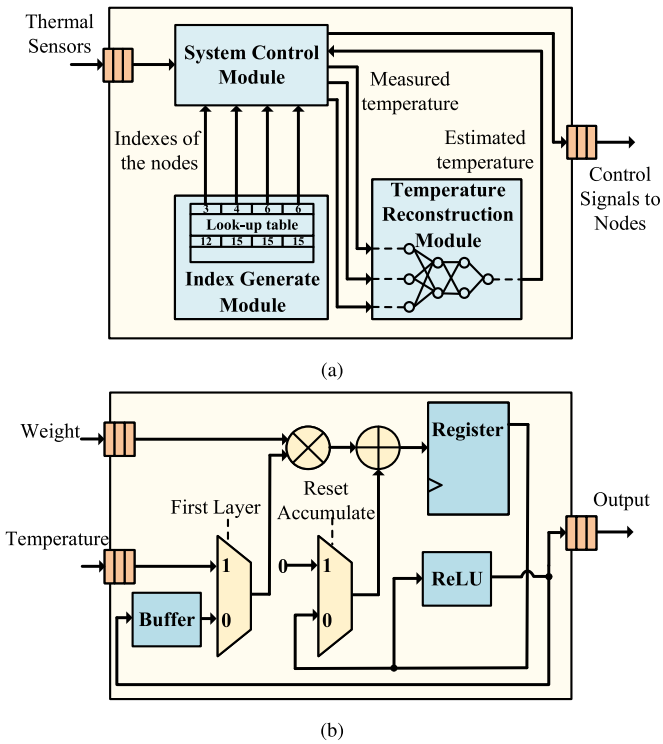


Fig. 5. (a) Schematic about how the temperature reconstruction unit works. (b) Hardware structure about implementing the ANN.

1) *Searching Sensors in Different Directions:* According to the Matérn correlation function, we have concluded that the correlation of two nodes is mainly dependent on the distance between them. The closer their distance, the more correlation they have. Therefore, we search for the nearest sensor in the X–Z directions for each node in the set $\mathcal{N}$. We first check whether there is a sensor-allocated node at a distance of one grid in a certain direction. If the neighbor node embeds a thermal sensor, we place the neighbor node's index into the lookup table. Otherwise, we will continue to search with the expanded search range. If the sensor is still not found when the search range exceeds the size of the NoC-based multicore system, -1 will be stored in the corresponding position in the lookup table, which means there is no sensor in this direction.
2) *Handling Data Hazards:* As introduced above, the number of input neurons is three. When the number of sensors available is small, there may be no sensor in one of the neighbor directions of the nonsensor-allocated nodes. It does not satisfy the input requirements of the neural network in this case, which is reflected in the lookup table as -1 for some elements. We handle the data hazards using the ring assignment method: a) assigning the element at Z to the invalid X; b) assigning the element at X to the invalid Y; and c) assigning the element at Y to the invalid Z. Using the ring assignment method, we guarantee enough input data for the ANN to work when the number of the available sensors is so small that there is no sensor allocated in some directions. In the ring assignment method, the invalid neighbor node's temperature in one direction is assumed to be the same as the one in another direction. This assumption is reasonable due to the thermal spatial locality.

Fig. 4 shows a simple example about how to build a lookup table for a 3 × 3 × 3 NoC-based multicore system. For core 0, an element of the set $\mathcal{N}$, core 1 and 2 are in its X direction. However, none of them have embedded thermal sensors. Thus, the X-id of core 0 is marked as -1. In the Y direction, there are core 3 and 6. We can see that $6 \in \mathcal{L}$ and $3 \notin \mathcal{L}$, so the Y-id is 6. Similarly, Z-id is 18. If there is more than one sensor with the same distance in the same direction, we randomly select one of them as the element in the lookup table. For example, for core 7, there are core 6 and 8, both embedding thermal sensors, in one grid away in the X direction. Therefore, we can randomly choose one from $\{6, 8\}$ as X-id.

After that, we use ring assignment to eliminate the invalid indexes -1. For core 12, with invalid indexes X-id and Z-id, we first assign Y-id to Z-id and then assign Z-id to X-id. The ring assignments scheme cannot work when there is no valid index. Therefore, we make this case taboo to prevent GA from generating such individuals.

V. Hardware Implementation of the Reconstruction Unit

The schematic about how our temperature reconstruction unit works is shown in Fig. 5(a). The reconstruction unit consists of a system control module, an index generate module, and a temperature reconstruction module. When estimating a nonsensor-allocated core's temperature, the index generation module first generates three indexes of the needed cores according to the lookup table and transmits them to the system control module, which stores the temperature of the sensor-allocated cores. The temperature reconstruction module uses the temperature measurements as input data of the ANN to

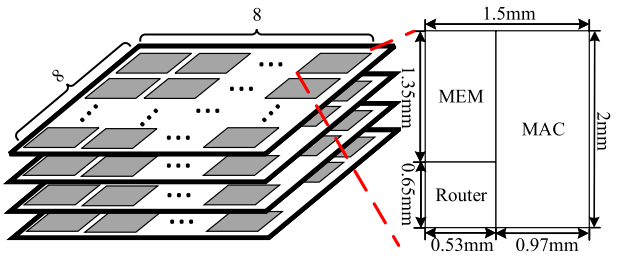


Fig. 6. Sketch map of the simulated NoC-based multicore system and the structure of the nodes.

TABLE I
CONFIGURATIONS OF THE 3-D NOC-BASED MULTICORE SYSTEM

Parameter	Specification	Parameter	Specification
Physical size (mm)			
Router Length	0.72	Router Width	0.65
MAC Length	2.00	MAC Width	0.78
Memory Length	1.35	Memory Width	0.72
Thickness (mm)			
Silicon	0.15	Thrmal Interface Material	0.15
Specific Heat capacity ($J/(m^3K)$)			
Silicon	1.75×10^6	Thrmal interface material	4.0×10^6

estimate the core's temperature. The ANN output and the estimated core index will be restored in the system control module. As the temperature reconstruction unit is global, the above steps need to be performed multiple times for different nonsensor-allocated cores to reconstruct the full-chip temperature. Although the temperature data is transmitted over the shared on-chip network, it does not significantly impact the network traffic load because the full-chip temperature is only reconstructed once in a long period interval. For example, when the number of thermal sensors is 70 and the period interval is 100 000 cycles, only 70 temperature data are transmitted from thermal sensors to the global temperature reconstruction unit through the on-chip network in every 100 000 cycles. Therefore, the additional network traffic load is negligible.

The hardware structure of the ANN is described in Fig. 5(b). We reuse one multiplier and one accumulator for ANN layer computation to reduce the area consumption. When calculating the input layer, the temperature data transmitted from the system control model are selected as the input data. When the calculation of the current layer is completed, the results will be stored in the buffer. These results are selected as the input data when calculating the next layer.

Although only one multiplier and one accumulator are used, the computing latency is acceptable for the real-time system. The size of the ANN we used is $3 \times 2 \times 2 \times 1$. Thus, the hardware needs to execute 12 times of multiplication and addition to calculate one non-sensor-allocated node's temperature, which consumes 12 cycles. The time complexity of our method is O(n), where n is the number of the nonsensor-allocated nodes. Besides, the total computing latency is equal to the product of 12 and the number of nonsensor-allocated nodes in the NoC-based multicore system. For example, when 70 sensors are embedded in a 256-core 3-D NoC-based multicore system, the computing latency is 2.232 μs under 1 GHz, and it is much smaller than the interval of temperature sampling, which is usually ms level.

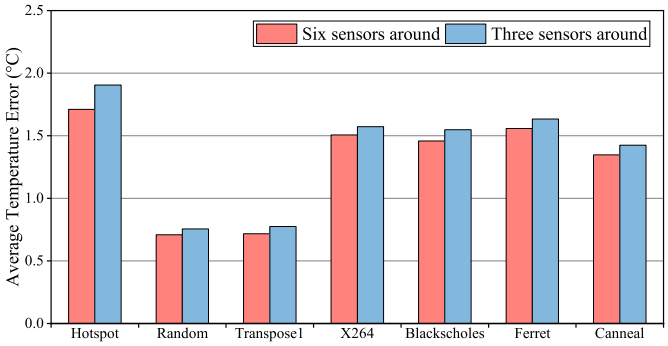


Fig. 7. Comparison of the estimation error between using six sensors and three sensors to surround the nonallocated nodes.

VI. EXPERIMENTAL RESULTS AND ANALYSIS

To obtain the temperature data for the ANN training and model evaluation, we implemented a $8 \times 8 \times 4$ mesh-based 3-D NoC-based multicore system on a cycle-accurate traffic-thermal NoC simulation tool-AccessNoxim [22]. Each node in the NoC-based multicore systems consists of a multiply-accumulate (MAC) block, a memory block, and a router block as shown in Fig. 6. Besides, there is a thermal interface material filling between the layers. The detailed physical configurations of the 3-D NoC-based multicore system are shown in Table I. The multicore system that presented in Fig. 6 and Table I is just a common case study, and it is used to study the effectiveness of the methods proposed in this article. The simulations are conducted with combinations of different injection rates, different routing algorithms such as ZXY, XYZ, West-First, North-Last, Fully-Adaptive, OddEven, and different traffic distributions such as Hotspot, Random, and Transpose1. The temperature data under these three traffic distributions are collected as the training data for sensor allocation and ANN training. Besides, we use Gem5 [23] to obtain the traffic traces of real applications, such as the princeton application repository for shared-memory computers (PARSEC) benchmarks [24]: 1) X264; 2) Blackscholes; 3) Ferret; and 4) Canneal, and then we apply those traces to AccessNoxim to get the thermal distribution. These four real applications' temperature data are unknown to the offline training process, which are used to test whether the reconstruction scheme is flexible.

A. Six Sensors Around or Three Sensors Around

In Section III-A, we propose using three sensors instead of six sensors to decrease the number of sensors and the cost. As shown in Fig. 7, using six sensors has a slight improvement in accuracy, about 6.33% decreasing on the average temperature error. This minor improvement is negligible compared to huge expenses.

B. Robustness

The initialization algorithm will give a different initial allocation each time. To evaluate the robustness of the nearest-neighbor-based initialization algorithm, we set $N_{\min} = 80$,

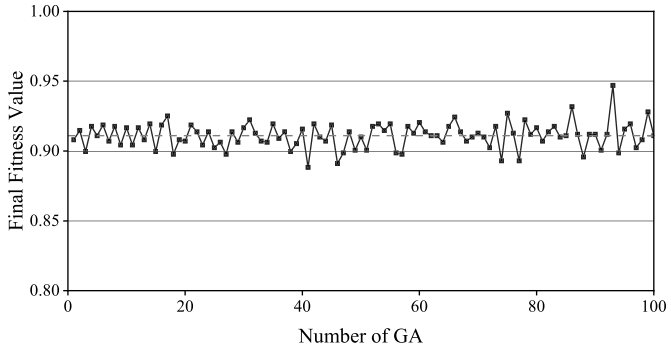


Fig. 8. Fluctuation of the fitness values of the optimal individuals in 100 GA optimizations.

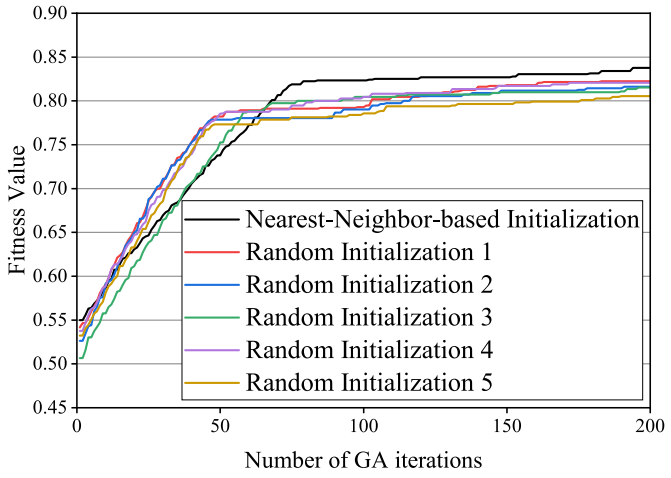


Fig. 9. Comparison of the fitness values between using Algorithm 1 and the random allocation method.

$N_{\max} = 110$, and execute the nearest-neighbor-based initialization algorithm and GA for 100 times. The fitness values are shown in Fig. 8. We use the coefficient of variation to describe the robustness. The coefficient of variation is defined as follows:

$$\text{cov} = \frac{\sigma}{\mu} \tag{6}$$

where σ is the standard deviation, and μ is the average value. The coefficient of variation is 0.06 for the 100 times, which indicates the method is robust, and the randomness in the initialization algorithm does not affect the final sensor allocation obtained by GA.

C. Optimizing Search

Our initialization algorithm is proposed to reduce the search space and enable the GA to find the optimal solution quickly. We compare the GA's search process using this initialization algorithm and a random initialization method to prove that our initialization algorithm can make the GA converge to a better solution consuming the same time. The random method will randomly allocate the same number of sensors as the proposed initialization method. As shown in Fig. 9, our method can make GA find an allocation with a higher fitness value after the convergence of GA.

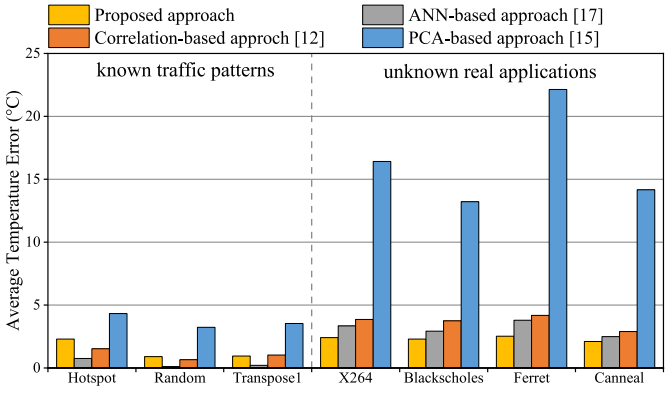


Fig. 10. Average temperature error under known traffic patterns and unknown real applications using the proposed approach, Co2-ANN-based approach [19], correlation-based approach [12], and PCA-based approach [15].

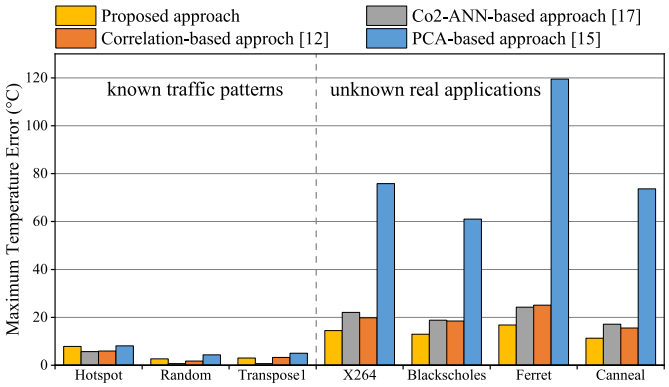


Fig. 11. Maximum temperature error under known traffic patterns and unknown real applications using the proposed approach, Co2-ANN-based approach [19], correlation-based approach [12], and PCA-based approach [15].

D. Average and Maximum Temperature Error

After training, we estimate the full-chip temperature of three different traffic patterns and four real applications, respectively, and calculate the error between the estimated and actual temperatures. We consider these three traffic patterns in the offline training phase, but the real applications have not appeared in the training data set. We use 70 sensors which account for 20.34% of the total cores. Furthermore, to decrease the impact caused by randomness in the initial allocation, we repeated the experiments ten times. We used the average results under these ten times to plot Figs. 10 and 11. As shown in Fig. 10, the average temperature error of our approach is equivalent to [12] and better than [15] when it comes to the known traffic patterns. For the real applications, our approach reduces 17.60%–88.63% average error compared with the state-of-the-art methods [12], [15], [19]. As shown in Fig. 11, the maximum temperature error of our approach is equivalent to [12] and better than [15] under known traffic patterns. For the real applications, our approach reduces 26.97%–85.92% maximum temperature error compared with the state-of-the-art methods [12], [15], [19]. The ability to track thermal hotspots determines the maximum

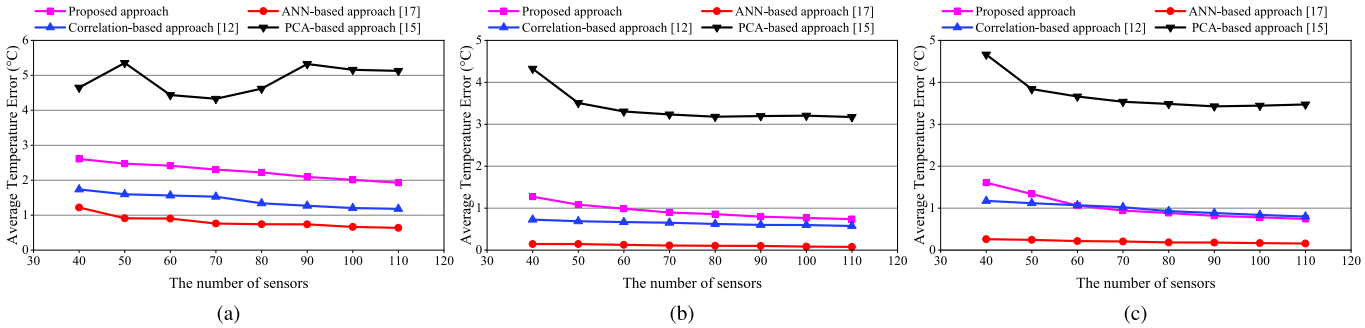


Fig. 12. Average temperature error between the estimated temperature and the true temperature using the proposed approach, Co2-ANN-based approach [19], correlation-based approach [12], and PCA-based approach [15] under different known traffic patterns. (a) Hotspot. (b) Random. (c) Transpose1.

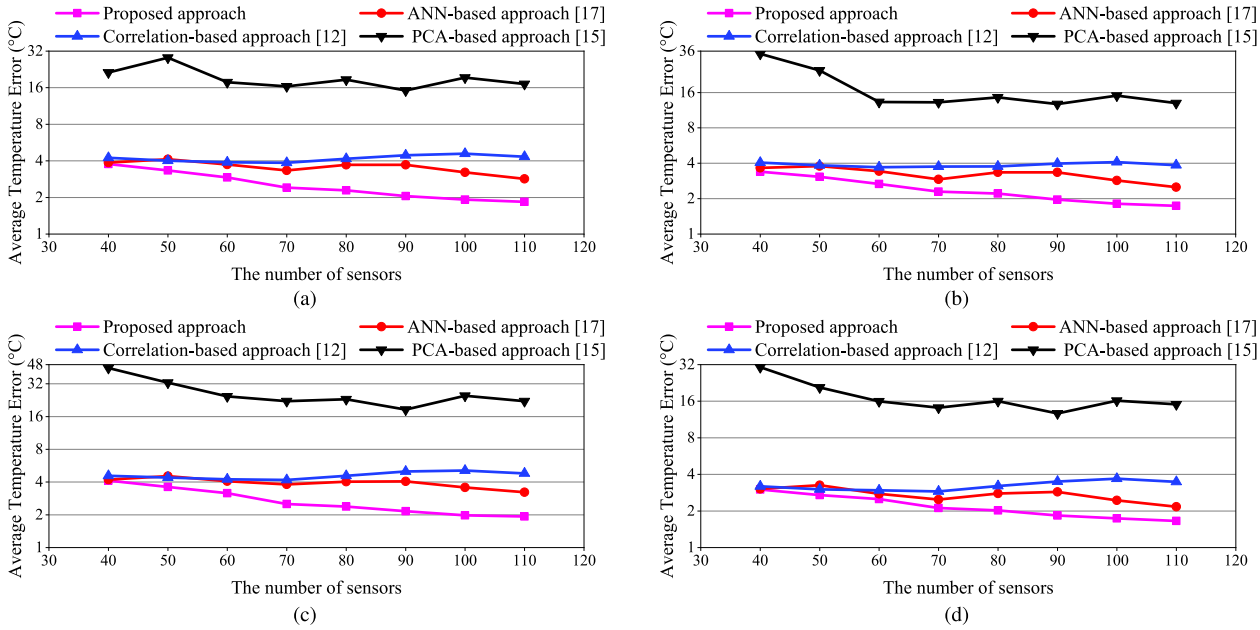


Fig. 13. Average temperature error between the estimated temperature and the true temperature using the proposed approach, Co2-ANN-based approach [19], correlation-based approach [12], and PCA-based approach [15] under different unknown real applications. (a) X264. (b) Blackscholes. (c) Ferret. (d) Canneal.

temperature error. The Co2-ANN-based method [19] has the highest accuracy under known traffic patterns due to the powerful learning ability of its complex neural network. However, its hardware cost is high. For the unknown real applications, our proposed method has the minimum error between these methods. This is because methods [12], [15], [19] depend on the training data. When the work scenarios differ from the offline training dataset, the cores' thermal relationship used in these methods will change. However, our sensor allocation algorithm is based on the spatial correlation, and this relationship does not vary as the application scenario changes.

E. Stability

To verify the stability of our reconstruction method, we change the number of thermal sensors within the range of 40–110, and the step is ten. In addition, we repeated ten experiments for each number of thermal sensors to reduce the impact of randomness in the initial allocation. The average results of these ten experiments are shown in Figs. 12 and 13. We still analyze the experimental results in two parts. For the known traffic pattern, our previous work [19] always has the best performance due to its large-scale neural network, but it leads to huge hardware overhead. In addition, our proposed method in this article is equivalent to [12] and is better than [15], which is consistent with the previous conclusion. For the unknown real applications, which are the more realistic scenarios, our method always has the lowest average temperature error with the different numbers of sensors. Besides, more thermal sensors cause less temperature error in our proposed approach. Meanwhile, the decrease of the average temperature error of [12], [15], and [19] is insignificant as the number of sensors rises, and the error even increases sometimes. This is because their sensor allocation algorithms are totally based on the known data. And the temperature relationship between the nodes learned by these methods does not work for the application scenarios with different characteristics from the known dataset.

TABLE II
COMPARISON ABOUT THE HARDWARE COST

Approach	Logic area (μm^2)	Memory size (Kb)
Proposed approach	1179.74	6.19
Co2-ANN-based approach [19]	8076.60	356.56
Correlation-based approach [12]	544.95	10.00
PCA-based approach [15]	3278.77	240.00

F. Hardware Cost

Under 70 thermal sensors, we calculate the memory size needed to store the weights and the table that describes the correlation relationship between the nodes. Besides, we obtain the logic hardware overhead for these four approaches using Synopsys Design Compiler (DC) under TSMC 28 nm technology and 1 GHz frequency. The result is shown in Table II. The approach [12] only uses a multiplier and an adder to build their linear-regression-based temperature reconstruction module. Thus, its logic area is small. The area of our approach is larger than [12] due to the control logic in ANN. The approaches in [15] and [19] use more adders and multipliers, which cause a large area. For the memory size, considering the lookup table and the weights of the ANN, our method needs the fewest memory resources due to the small network. The approaches in [15] and [19] need a large memory due to their large number of weights.

VII. CONCLUSION

The thermal problem is serious in NoC-based multicore systems, especially when 3-D NoC-based multicore systems appear. To prevent overheating, the thermal sensors are usually embedded in the system to get the temperature for control, but the number is always limited due to the cost. In this article, we propose a thermal sensor allocation method based on the spatial correlation and GA. Besides, to reconstruct the full-chip temperature, we use ANN instead of the linear methods to learn the correlation information between the non-sensor-allocated nodes and the sensor-allocated nodes. The experimental results show that our method has outstanding flexibility, and it can work under various scenarios with high accuracy even though the scenarios do not appear in the offline phase. Compared with the conventional approach, our approach reduces 17.60%–88.63% average temperature error for the unknown real applications. For the hardware overhead, our approach is equivalent to [12] and is less than [15] and [19].

REFERENCES

[1] R. H. Dennard, F. H. Gaensslen, V. L. Rideout, E. Bassous, and A. R. LeBlanc, "Design of ion-implanted MOSFET's with very small physical dimensions," *IEEE J. Solid-State Circuits*, vol. SSC-9, no. 5, pp. 256–268, Oct. 1974.

[2] J. Wang and Y. Ye, "Ant colony optimization-based thermal-aware adaptive routing mechanism for optical NoCs," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 40, no. 9, pp. 1836–1849, Sep. 2021.

[3] W. J. Dally and B. Towles, "Route packets, not wires: On-chip interconnection networks," in *Proc. 38th Design Autom. Conf.*, Jun. 2001, pp. 684–689.

[4] L. Benini and G. De Micheli, "Networks on chips: A new SoC paradigm," *Computer*, vol. 35, no. 1, pp. 70–78, Jan. 2002.

[5] N. Shahabinejad and H. Beitollahi, "Q-thermal: A Q-learning-based thermal-aware routing algorithm for 3-D network on-chips," *IEEE Trans. Compon., Packag., Manuf. Technol.*, vol. 10, no. 9, pp. 1482–1490, Sep. 2020.

[6] R. Joseph, L. Shang, R. P. Dick, and D. Brooks, "Power, thermal, and reliability modeling in nanometer-scale microprocessors," *IEEE Micro*, vol. 27, no. 3, pp. 49–62, May 2007.

[7] K. N. Dang, A. B. Ahmed, A. B. Abdallah, and X.-T. Tran, "HotCluster: A thermal-aware defect recovery method for through-silicon-vias toward reliable 3-D ICs systems," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 41, no. 4, pp. 799–812, Apr. 2022.

[8] A. K. Coskun, J. L. Ayala, D. Atienza, T. S. Rosing, and Y. Leblebici, "Dynamic thermal management in 3D multicore architectures," in *Proc. Design, Autom. Test Eur. Conf. Exhib.*, Apr. 2009, pp. 1410–1415.

[9] Y. Fu, L. Li, K. Wang, and C. Zhang, "Kalman predictor-based proactive dynamic thermal management for 3-D NoC systems with noisy thermal sensors," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 36, no. 11, pp. 1869–1882, Nov. 2017.

[10] S. Rahimipour, W. N. Flayyih, N. A. Kamsani, S. J. Hashim, M. R. Stan, and F. Z. B. Rokhani, "Low-power, highly reliable dynamic thermal management by exploiting approximate computing," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 28, no. 10, pp. 2210–2222, Oct. 2020.

[11] K.-C.-J. Chen and Y.-H. Liao, "Adaptive machine learning-based temperature prediction scheme for thermal-aware NoC system," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Oct. 2020, pp. 1–4.

[12] K.-C. Chen, H.-W. Tang, Y.-H. Liao, and Y.-C. Yang, "Temperature tracking and management with number-limited thermal sensors for thermal-aware NoC systems," *IEEE Sensors J.*, vol. 20, no. 21, pp. 13018–13028, Nov. 2020.

[13] S. Reda, R. Cochran, and A. N. Nowroz, "Improved thermal tracking for processors using hard and soft sensor allocation techniques," *IEEE Trans. Comput.*, vol. 60, no. 6, pp. 841–851, Jun. 2011.

[14] A. N. Nowroz, R. Cochran, and S. Reda, "Thermal monitoring of real processors: Techniques for sensor allocation and full characterization," in *Proc. Design Automat. Conf. (DAC)*, Jun. 2010, pp. 56–61.

[15] J. Ranieri, A. Vincenzi, A. Chebira, D. Atienza, and M. Vetterli, "Near-optimal thermal monitoring framework for many-core systems-on-chip," *IEEE Trans. Comput.*, vol. 64, no. 11, pp. 3197–3209, Nov. 2015.

[16] K.-C. Chen, Y.-H. Chen, and Y.-P. Lin, "Thermal sensor allocation and full-system temperature characterization for thermal-aware mesh-based NoC system by using compressive sensing technique," in *Proc. Int. Symp. VLSI Design, Autom. Test (VLSI-DAT)*, Apr. 2017, pp. 1–4.

[17] K.-C. Chen, H.-W. Tang, C.-H. Wu, and C.-H. Chen, "Thermal sensor placement for multi-core systems based on low-complex compressive sensing theory," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 41, no. 11, pp. 5100–5111, Nov. 2022.

[18] X. Li, Z. Li, X. Ou, Y. Liu, W. Zhou, and Z. Duan, "High-resolution thermal maps extraction of multi-core processors based on convolutional neural networks," in *Proc. 45th Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Oct. 2019, pp. 3075–3080.

[19] M. Guo, T. Cheng, L. Li, and Y. Fu, "Optimized method for thermal tracking in 3D NoC systems by using ANN," in *Proc. 18th Int. SoC Design Conf. (ISOCC)*, Oct. 2021, pp. 111–112.

[20] B. Hargreaves, H. Hult, and S. Reda, "Within-die process variations: How accurately can they be statistically modeled?" in *Proc. Asia South Pacific Design Autom. Conf.*, Jan. 2008, pp. 524–530.

[21] J. Ranieri, A. Chebira, and M. Vetterli, "Near-optimal sensor placement for linear inverse problems," *IEEE Trans. Signal Process.*, vol. 62, no. 5, pp. 1135–1146, Mar. 2014.

[22] K.-Y. Jheng, C.-H. Chao, H.-Y. Wang, and A.-Y. Wu, "Traffic-thermal mutual-coupling co-simulation platform for three-dimensional network-on-chip," in *Proc. Int. Symp. VLSI Design, Autom. Test*, Apr. 2010, pp. 135–138.

[23] N. Binkert et al., "The gem5 simulator," *ACM SIGARCH Comput. Archit. News*, vol. 39, no. 2, pp. 1–7, Aug. 2011, doi: 10.1145/2024716.2024718.

[24] C. Bienia, "Benchmarking modern multiprocessors," Ph.D. dissertation, Dept. Comput. Sci., Princeton Univ., Princeton, NJ, USA, 2011.

Menghao Guo received the B.E. degree from the School of Electronic Science and Engineering, Nanjing University, Nanjing, China, in 2022. He is currently pursuing the M.E. degree with the School of Integrated Circuits, Tsinghua University, Beijing, China.

His current research interests include thermal tracking in multicore systems and computer architecture.

Tong Cheng received the B.E. degree from the School of Electronic Science and Engineering, Nanjing University, Nanjing, China, in 2022, where he is currently pursuing the M.E. degree with the School of Electronic Science and Engineering.

His research interests include network-on-chip (NoC) design and thermal management in NoC.

Xinyi Li is pursuing the B.E. degree in communication engineering with Nanjing University, Nanjing, China.

Her current research interests include network-on-chip (NoC) design and thermal management in NoC.

Li Li (Member, IEEE) received the B.S. and Ph.D. degrees from the Hefei University of Technology, Hefei, China, in 1996 and 2002, respectively.

She is a Professor of the School of Electronic Science and Engineering, VLSI Design Institute, Nanjing University, Nanjing, China. Her current research interests include VLSI design for digital signal processing systems reconfigurable computing and multiprocessor system-on-chip (MPSoC) architecture design methodology.

Dr. Li is a member of the Circuits and Systems for Communications (CASCOM) TC of IEEE CAS Society.

Yuxiang Fu (Member, IEEE) received the B.S. degree in microelectronics and solid state electronics and the Ph.D. degree in electronic science and technology from Nanjing University, Nanjing, China, in 2013 and 2018, respectively.

In 2018, he joined the School of Electronic Science and Engineering, Nanjing University. Now, he is an Assistant Professor with the School of Integrated Circuits, Nanjing University. His current research interests include AI for chip architecture design, network-on-chip algorithms/architectures, low-power digital systems, and 3-D IC design.