



A Heterogeneous Multi-core Network-on-Chip Mapping Optimization Algorithm

Juan Fang^(✉), Haoyan Zhao, Jiayue Zhang, and Jiamei shi

Beijing University of Technology, Beijing 100124, China
{fangjuan, zhangjiayue}@bjut.edu.cn

Abstract. Heterogeneous multi-core Network-on-Chip provides high-speed execution and performance to meet the heavy communication demands of the system. However, the system power consumption and delay increase as the number of cores rises. In this paper, we present KL_SA, a mapping scheme based on the Kernighan-Lin (KL) and Simulated Annealing (SA) algorithms, aiming at the mapping optimization problem on heterogeneous multi-core Network-on-Chip. This scheme combines the advantages of the KL algorithm's efficient partitioning and the SA algorithm's global search for optimal solution and improves the latter in turn. Firstly, we divide the task using KL algorithm, with which result to initial the mapping of the SA algorithm, solving the random initialization problem in the SA algorithm, also increasing the likelihood of getting the optimal solution. Secondly, we do the mapping using the SA algorithm with memory function added in the iterative process. In this way, the current best state is memorized without losing the current optimal solution when escaping from the local optimum, ensuring the global optimal approximate solution obtained. Experiments show large savings in the aspects of system power consumption and delay on the system with the proposed mapping scheme compared to the existing mapping schemes.

Keywords: Network on chip · KL_SA algorithm · Mapping optimization · Global optimal approximate solution

1 Introduction

The emergence of Network-on-Chip (NoC) enables numerous processing units (also known as IP cores) being integrated so as to provide a higher system performance. While NoC contributes to a higher system performance with its speed-up execution, it also brings the problem of energy consumption growth and delay increment. This problem may reduce the life span of chip and affect the stability of chip's operation due to the limitation of chip area and heat dissipation capacity, which limits further improvement of system performance. Thus, it is crucial to improve the energy efficiency in NoC design.

In previous works, low-power, high-performance topology and routing algorithms [1–3] have been designed to lower the NoC energy consumption. In recent years, the study on NoC mapping algorithm with power consumption or delay as the optimization

target has gained extensive attention from scholars and designers globally. Since the communication cost among the NoC nodes is largely affected by the communication distances between the nodes of NoC, it is desirable to design an effective NoC mapping algorithm to reduce the average distance between nodes.

The task mapping problem is generally known to be a NP-hard problem [17]. The widely-used method to solve this kind of problem in a reasonable amount of time is to obtain near-optimal solutions using meta-heuristic algorithms, e.g. simulated annealing [13–15]. Studies have shown that implementing an effective NoC mapping algorithm can significantly reduce the overall power consumption of the system, and improve system performance [4–6].

In addition, various analysis and heuristic algorithms for mapping optimization [7–12] have been proposed. Alagarsamy et al. [13] proposed a mapping scheme with the meta-heuristic search algorithm of SA and Tabu search (SAT) to analyze and optimize the power consumption based on NoC system. Jiang et al. [16] devised a new fault model based on the standby core technology, and designed a fault-recognition low-power dynamic task mapping algorithm targeting on NoC system fault, power consumption and system performance. Taassori et al. [17] proposed a linearized Quadratic Allocation Problem (QAP) model, mapping tasks to the kernel, which not only minimized the power consumption, but also improved the performance of NoC. Le et al. [18] designed a mapping optimization algorithm based on distributed search (SS) and improved multi-objective optimization of standard SS, which helped to obtain high-performance mapping layout.

Inspired by the aforementioned studies of NoC mapping algorithm, we discover a pattern in NoC mapping for effectively reducing communication cost among NoC nodes. We then analyze the pattern and explore the principle behind it. Instead of mapping IP cores to NoC nodes directly, we break it into two phases. The first phase is to assign the tasks to IP cores according to the relatedness between the tasks. The second phase is the mapping between the IP cores and the NoC routing nodes. In this way, the more related tasks are assigned to one IP core, then, the relatedness between IP cores is reduced, thereby, the interactions between IP cores are cut down. This means that after the IP cores are mapped to NoC nodes in the second phase, the locality of data communication among NoC nodes becomes higher, which leads to a shorter average communication distance and therefore a lower energy consumption.

Hence, how to divide the tasks according to their relatedness and then how to integrate the two phases leaves us the space to innovate. The key is to find an appropriate algorithm for each phase and combine the two with taking the advantage of them. The KL algorithm [20] has the advantage of efficient partitioning, which can quickly divide the whole space into multiple subspaces. On the other hand, the SA algorithm [21] can avoid the premature phenomenon better by introducing the Metropolis criterion, and has the advantage of global search optimal solution, which can find the solution that is nearest to the optimal solution in the whole solution space.

The contribution of this paper is three-fold. Firstly, we discover a pattern for NoC mapping to reduce energy consumption and analyze the principle behind the pattern. Secondly, we propose a new mapping scheme to reduce power consumption and delay for the heterogeneous multi-core NoC, and the proposed scheme can achieve global optimal

approximate solution with less complexity than the existing mapping schemes proved by abundant experimental results. Finally, our proposed scheme provides a new perspective for heterogeneous multi-objective optimization problem by utilizing the partitioning of KL algorithm to remedy the drawback of SA algorithm’s random initialization.

In the following sections, we first describe the NoC mapping problem and the mapping optimization model in Sect. 2, then we present the implementation of the proposed scheme in Sect. 3, finally we show the evaluation and analysis in Sect. 4, and conclusion is in Sect. 5.

2 NoC Mapping Problem Description and Mapping Optimization Model

2.1 Overview

In a heterogeneous environment, this paper divides the mapping into two phases, as shown in Fig. 1. In the first phase, the task nodes in the task graph are assigned to the appropriate IP cores. In the second phase, the IP cores that have been bound to the task are mapped to the NoC platform to settle their specific locations in the NoC. T represents the subtask in the application, IP represents the IP core, and R represents the NoC routing node.

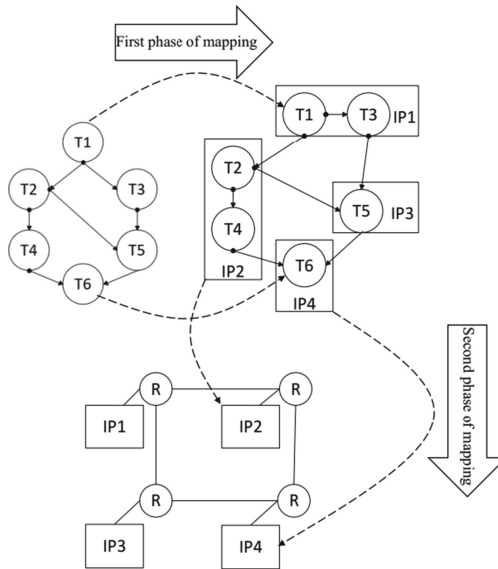


Fig. 1. Two phases of NoC mapping

The NoC used in the study of the mapping scheme in this paper is a 2D Mesh topology, as shown in Fig. 1, where each resource node consists of a processing module (IP) and a routing node (R), which perform data interaction through a network interface (NI). As

for the route assignment in NoC mapping, the routing algorithm of static shortest path is applied in our NoC platform. It is because it is simple to understand and implement, and the conclusion can be easily extended to other more complicated NoC architecture.

2.2 Problem Definition

In this section, the mathematical description of the NoC mapping problem is presented. First, we give the following three basic definitions.

Definition 1: The task graph is a directed graph, denoted as $TG(T, R, V)$, where each vertex $T_i \in T$ represents a subtask, and each directed edge $R_{i,j} \in R$ represents a communication relationship from subtask T_i to T_j , and the edge weight $V_{i,j}$ represents the traffic of subtasks T_i to T_j .

Definition 2: The given core communication graph is a directed graph, denoted as $CG(C, R, V)$, where each vertex $C_i \in C$ represents an IP core, and each directed edge $R_{i,j} \in R$ represents a communication relationship from the IP core C_i to C_j , and the edge weight $V_{i,j}$ represents the traffic of IP core C_i to C_j .

Definition 3: The given NoC platform topology graph is a directed graph, denoted as $NG(N, P, E)$, where each vertex $N_i \in N$ represents a resource node in the NoC, $P_{i,j} \in P$ represents the routing path from network node N_i to N_j , and $E_{i,j}$ represents the communication cost by transmitting 1 bit of data from network node N_i to N_j .

Based on the above definitions, the mathematical description of the mapping problem for power consumption and delay is as follows.

2.3 NoC Power Consumption Model

The total energy consumed in the entire NoC system consists of the energy consumed by the IP core E_c in the processing task and the energy consumed during the data communication process E_{net} , as shown in the following formula.

$$\text{Energy} = E_c + E_{net} \quad (1)$$

The energy consumed by the IP core in processing the task is only related to the type of the core, so the processing power consumption of the v bits data on a specific core is:

$$E_{ip} = \rho_e * v \quad (2)$$

where ρ_e is a constant associated with the type of IP core.

So, the total power consumption on the IP core is:

$$E_c = \sum_{ip=1}^m E_{ip} \quad (3)$$

where m represents the number of IP core.

For the power consumption in data communication, the power consumption during single bit data transmission is:

$$E_b = E_s + E_a + E_w + E_l \quad (4)$$

where E_b represents the power consumption generated by transferring unit data from one network node to another network node, E_s represents the power consumption generated by the crossbar switch in the router, E_a represents the power consumption generated by the unit data in the internal buffer area of the routing node, E_w represents the power consumption generated by the internal wiring of the router, and E_l represents the power consumption generated by the unit data through the NoC communication interconnection link. The values of E_s , E_a and E_w are mainly related to the internal design of the routing node, and will not change with the communication status in the network. It can be approximated as a constant, so we use E_r to represent it uniformly. Then the power consumption generated during unit data transmission can be expressed by Formula (5).

$$E_b^{i,j} = n_{i,j} * E_r + (n_{i,j} - 1) * E_l \quad (5)$$

Where $E_b^{i,j}$ is the power consumption generated by transmitting unit data from the network node N_i to the network node N_j , and $n_{i,j}$ is the number of network nodes through which the unit data passes during transmission. Therefore, the power consumption generated by communication between node T_i and node T_j can be calculated by the Formula (6).

$$E_{i,j} = v_{i,j} * E_b^{i,j} \quad (6)$$

Where $v_{i,j}$ represents the amount of communication between network nodes N_i and N_j .

Therefore, the total power consumption in data communication can be calculated by the Formula (7).

$$E_{\text{net}} = \sum_{i=1}^n \sum_{j=1}^n E_{i,j} \quad (7)$$

Where n represents the number of NoC node.

Thus, the power consumption in the entire NoC system can be obtained from the Formula (8).

$$\text{Energy} = \sum_{ip=1}^m E_{ip} + \sum_{i=1}^n \sum_{j=1}^n E_{i,j} \quad (8)$$

2.4 NoC Delay Model

The delay in the entire NoC system mainly comes from the time consumption when the IP core processes the data D_c and the time consumption of the data during the NoC transmission D_{net} .

The time consumption of the IP core in processing the data is only related to the type of the core, so the time consumption of the v bits data on a specific core is:

$$D_c = \rho_d * v \quad (9)$$

where ρ_d is a constant associated with the type of IP core.

The time consumption of data in the NoC transmission mainly includes three parts: the delay on the source and sink nodes NI, the delay generated by the routing node during transmission, and the delay of the transmission path interconnection line. Therefore, the time consumption during unit bit data transmission can be expressed as formula (10).

$$D_b = 2 * \rho_N + \rho_R + \rho_L \quad (10)$$

Where ρ_N , ρ_R , and ρ_L are the constants of the transmission time on the NI, router, and interconnect lines.

With the above formula, we can get the time consumption of unit data transmission from network node N_i to network node N_j :

$$D_b^{i,j} = 2 * n_{i,j} * \rho_N + n_{i,j} * \rho_R + (n_{i,j} - 1) * \rho_L \quad (11)$$

where $n_{i,j}$ is the number of network nodes through which the unit data passes during the transmission. So, the time consumption of communication between node N_i and node N_j can be calculated by the following formula:

$$D_{net} = v_{i,j} * D_b^{i,j} \quad (12)$$

where $v_{i,j}$ represents the amount of communication between network nodes N_i and N_j .

Therefore, we can calculate the delay in the entire NoC system based on the time consumption when the IP core processes the data D_c , the time consumption of the data during the NoC transmission D_{net} , and the dependencies between the tasks. The formula is as follows:

$$Delay = D_c + D_{net} \quad (13)$$

2.5 NoC Multi-objective Optimization Model

As the size of the NoC and the complexity of the application increases, the optimization goal of the mapping is no longer limited to a single target such as power consumption, delay or heat balance. The multi-objective optimization is becoming more and more needed. This paper establishes an evaluation function that weighs two optimization goals for the two targets of power consumption and delay simultaneously.

The existing evaluation functions designed for multi-objective optimization mainly include linear weighted summation method, square weighted summation method, ideal point method and other common methods [20]. In this paper, the linear weighted summation method is used to establish the multi-objective optimization evaluation function. For the multi-objective optimization problem of power consumption and delay, the weights

α_e and α_d are determined according to the importance of power consumption and delay, and subjected to the following conditions: $\alpha_e \geq 0$, $\alpha_d \geq 0$, $\alpha_e + \alpha_d = 1$. Therefore, the evaluation function is defined as follows:

$$\text{Fevaluation} = \alpha_e * \text{Energy}(M) + \alpha_d * \text{Delay}(M) \quad (14)$$

where $M \in \text{MAP}$, MAP is the set of all mapping schemes. After the weights α_e and α_d are set, different evaluation values Fevaluation are obtained by changing the mapping scheme M , and the smaller the value of Fevaluation is, the lower the power consumption and delay obtained by the current mapping can be.

3 Implementation of Mapping Scheme Based on KL_SA Algorithm

3.1 NoC Multi-objective Optimization Model

This paper proposes the KL_SA algorithm, which combines the advantages of the KL algorithm and the SA algorithm and improves the latter. The KL_SA algorithm has the advantage of efficient and global search for optimal solution. By using the mapping scheme generated by the KLc_SA algorithm, the communication distance between nodes with large traffic can be effectively reduced, and the solution closest to the optimal solution can be quickly obtained, thereby it can reduce the power and delay of the system.

3.2 NoC Mapping Process Based on KL_SA Algorithm

The mapping of task to IP core: Because the power consumption and delay estimated at this phase does not involve the communication distance, it is only related to the IP core type. Therefore, the basic SA algorithm is adopted as the mapping scheme in this phase. The specific steps are as follows.

1. Randomly assign the subtasks in the task communication graph to the IP core, and initialize the algorithm parameters.
2. Calculate the evaluation value generated by the current allocation result (the result obtained by the evaluation function is called the evaluation value), and then perform the random disturbance, the disturbance mode is to exchange the subtasks of the two different positions with each other.
3. Calculate the evaluation value of the new allocation result. At the same time, a flag bit is added as the current optimal solution to record the optimal mapping result.
4. Calculate the difference between the old and new evaluation value. If the new evaluation value is better than the old evaluation value, replace the old evaluation value with the new evaluation value, otherwise accept the new mapping result with a certain probability. The purpose of this is that because continuing the search on the poorer mapping results may get a better solution, thereby jumping out of the local optimal solution and finding a more ideal solution. The smaller the difference between the current evaluation value and the new evaluation value, the lower the temperature will be and the closer the solution obtained is to the optimal solution. So the acceptance probability will decrease as the difference and temperature decrease.

5. Repeat steps 2 to 4 according to the number of iterations at initialization until pre-specified the number of iterations is reached, and then execute 6.
6. Check the temperature as it is dropping, if it reaches the stop temperature, stop the iteration and record the final distribution result, otherwise, return to step 2 again.

When the algorithm converges as the temperature finally meets the stop criteria, the mapping of the task to the IP core is completed.

The Mapping of IP Core to NoC Platform: The allocation result of the first stage is taken as the initial input, and the KL_SA algorithm is adopted as the NoC mapping scheme in this phase.

The specific steps are as follows.

1. The IP core set in the core communication graph is regarded as a set of nodes, which is recorded as set V .
2. The partitioning operation is performed on the set V , and the steps of dividing the operation are as follows.
 - (1) Randomly divide the set into two subsets A and B containing the same number of nodes.
 - (2) Calculate the internal communication cost and external communication cost of the subsets A and B respectively. The calculation formula is:

$$\begin{aligned}
 E_a &= \sum_{v_i \in A, v_j \in B} w(v_i, v_j) \\
 E_b &= \sum_{v_i \in B, v_j \in A} w(v_i, v_j) \\
 I_a &= \sum_{v_i \in A, v_j \in A} w(v_i, v_j) \\
 I_b &= \sum_{v_i \in B, v_j \in B} w(v_i, v_j)
 \end{aligned} \tag{15}$$

where I_a, I_b represents the internal communication cost of the set A, B , and E_a, E_b represents the external communication cost of the set A, B , v_i, v_j represents an IP core node, $w(v_i, v_j)$ represents the communication cost of nodes v_i to v_j .

- (3) The nodes in subsets A and B are exchanged until a partition is found, maximizing the difference between internal and external communication costs. The difference D of the communication cost is calculated by the Formula (16).

$$D = I_a + I_b - (E_a + E_b) \tag{16}$$

3. The generated new set is recursively divided until there are only 2 subtasks in each divided subset. Finally get a set of sets, each of which has two subtasks.

4. We use the final set as the IP core to the NoC node initialization mapping method, and then initialize the algorithm parameters. The IP cores are randomly allocated to the NoC of the unit of subset, and 2 nodes in each subset are to be allocated to adjacent positions of NoC, where each node uniquely corresponds to a NoC resource node.
5. Calculate the current mapping evaluation value E_o , and set a flag E_f to be the current optimal solution, initialized to E_o .
6. Perform random perturbation by interchanging the IP core mappings of two different locations.
7. Calculate the evaluation value of the new mapping E_n .
8. If the new evaluation value is better than the old evaluation value $E_n < E_o$, replace the old evaluation value with a new evaluation value $E_o = E_n$, if the new evaluation value is better than the current optimal solution $E_n < E_f$, update the current optimal solution to the new evaluation value $E_f = E_n$.
9. If the conditions in step 8 is not satisfied, accept the new mapping result with a certain probability. This is because if the new result is accepted, it may get a poor solution while the new mapping result may bring the capacity to jump out of the local optimal solution and obtain the global optimal solution. Between accepting and rejecting, the probability value for accepting the new solution is:

$$P = \exp(\Delta E/T) \quad (17)$$

where ΔE represents the difference between the two mapping results, and T represents the current temperature.

10. Repeat steps 6 to 9 according to the number of iterations at initialization until the number of iterations is reached, and then execute 11.
11. Check the temperature as it keeps dropping, if it reaches the stop temperature, stop the iteration and record the final mapping result, otherwise return to step 6.

When the algorithm converges as the temperature finally meets the stop criteria, the mapping of the IP core to the NoC platform is completed.

4 Evaluation

4.1 Experimental Environment and Algorithm Parameter

In this section, we explain how the experiment is designed and how the evaluation of the algorithm performance is done. Our experiments were performed in a 4×4 2D Mesh NoC structure, in which a static shortest path routing algorithm [21] is used as the routing algorithm. In order to get more accurate experimental simulation data, we use the BookSim simulation simulator. BookSim [22] is a widely used cyclic precision interconnect network simulator designed by the Stanford University nocs group.

In order to verify the performance of the mapping scheme proposed in this paper, three applications are used to analyze the scheme. The first example is the MPEG-4 decoding system, the second example is the VOPD application, and the third example is the MWD application. We assign the tasks of the three application instances to different IP cores such as DSP, graphics processor, RISC CPU, etc. The mapped NoC platform uses a 4×4 2D Mesh structure. The SA algorithm is used as the baseline to compare with the KL_SA algorithm proposed in this paper. The experiments and analysis are carried out on power consumption optimization model, delay optimization model, and multi-objective optimization model.

4.2 Comparative Experiment and Result Analysis

Figure 2 shows the power consumption statistics of three applications using SA algorithm and the KL_SA algorithm with power consumption optimization as a single target. It can be seen from the figure that the mapping scheme generated by the KL_SA algorithm proposed in this paper has a large optimization in power consumption compared with the SA algorithm. For the MPEG-4 decoding system, the mapping scheme generated by the KL_SA algorithm is reduced by 15.9% compared with the SA algorithm. For the VOPD application, the mapping scheme generated by the KL_SA algorithm is reduced by 11.7% compared with the SA algorithm. For MWD applications, the mapping scheme generated by the KL_SA algorithm is reduced by 17.5% compared to the SA algorithm.

Figure 3 shows the delay statistics of three applications using SA algorithm and the KL_SA algorithm with delay optimization as a single target. We use the delay of the SA algorithm as the benchmark for analysis. It can be seen from the figure that the mapping scheme generated by the KL_SA algorithm proposed in this paper has a large optimization in delay than the SA algorithm. For the MPEG-4 decoding system, the mapping scheme generated by the KL_SA algorithm is reduced by 10.5% compared with the SA algorithm. For the VOPD application, the mapping scheme generated by the KL_SA algorithm is reduced by 9.6% compared with the SA algorithm. For MWD application, the mapping scheme generated by the KL_SA algorithm is 13.3% lower than the SA algorithm.

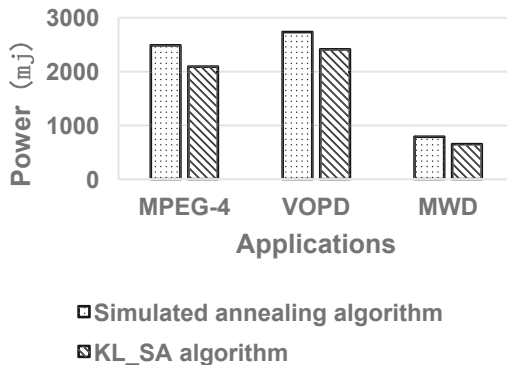


Fig. 2. Power consumption obtained by SA algorithm and KL_SA algorithm

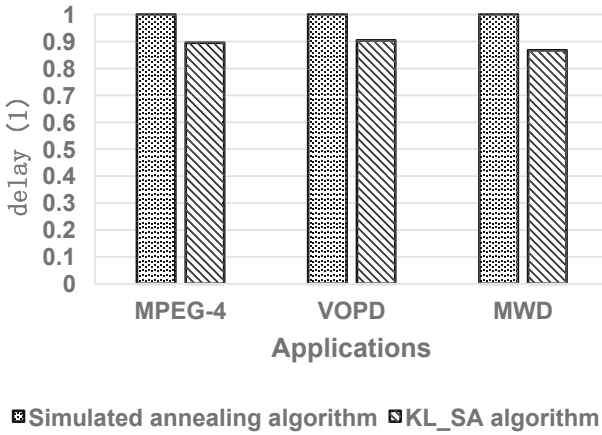


Fig. 3. Delay obtained by SA algorithm and KL_SA algorithm

For the multi-objective optimization of NoC, we performed set of experiments. we set the power consumption and delay weights to $\alpha_e = \alpha_d = 0.5$, Fig. 4(a, b) show the power consumption and delay statistics of the three applications using the SA algorithm and the KL_SA algorithm. We use the power consumption and delay of the SA algorithm as the benchmark. It can be seen from the figure that the mapping scheme generated by the KL_SA algorithm proposed in this paper has a large optimization in power consumption and delay compared with the SA algorithm. For the MPEG-4 decoding system, the mapping scheme generated by the KL_SA algorithm is 10.8% lower in power consumption and 8.5% lower in delay than the simulated annealing algorithm. For VOPD application, the mapping scheme generated by the KL_SA algorithm is better than the SA algorithm. As a result, the power consumption is reduced by 8.7% and the delay is reduced by 6.6%. For the MWD application, the mapping scheme generated by the KL_SA algorithm is 13.7% lower than the SA algorithm in power consumption, and the delay is reduced by 10.3%.

When adjusting the power consumption and delay weight α , the power consumption and delay optimization effects also change. Thus, we do more experiments with adjusting the value of α . First, we set $\alpha_e + \alpha_d = 1$, $\alpha_e = 0$, and then gradually increase the value of α_e until $\alpha_e = 1$. Figure 5(a, b) (where the abscissa represents the value of α_e , the three curves represent three different applications respectively) show the power consumption and delay optimization percentage respectively obtained by comparing the mapping scheme generated by the KL_SA algorithm proposed in this paper with the SA algorithm. When the weight α_e of the power consumption is gradually increasing, the power consumption optimization effect is more obvious, but the delay optimization effect is gradually weakened on the contrary. Therefore, we can adjust the value of α according to the needs of the application to get the best mapping results.

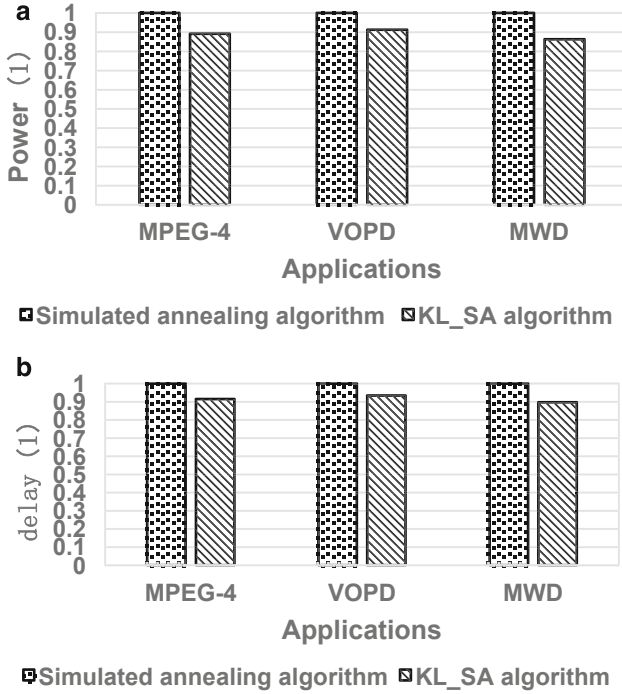


Fig. 4. (a) Power consumption obtained by SA algorithm and KL_SA algorithm (Multi-objective optimization, $\alpha_e : \alpha_d = 1 : 1$). (b) Delay obtained by SA algorithm and KL_SA algorithm (Multi-objective optimization, $\alpha_e : \alpha_d = 1 : 1$).

In order to better illustrate the effectiveness of the proposed scheme, we compare it with the methods in Refs. [23, 24] respectively. For the MPEG-4 decoding system, compared with the SA algorithm, the power consumption of the mapping scheme generated by the KL_SA algorithm proposed in this paper is reduced by 15.9%, and the power consumption of the mapping scheme proposed in the Ref. [23] is reduced by 6.7%. For the VOPD application, the power consumption of the mapping scheme generated by the KL_SA algorithm proposed in this paper is reduced by 11.3%, and the power consumption of the mapping scheme proposed in the Ref. [24] is reduced by 6.2%. Therefore, it can be seen that the mapping scheme proposed in this paper has better optimization effect.

Through experiments and results analysis, under the single-objective optimization condition, the power consumption and delay of the system are optimized by using the NoC mapping scheme generated by the KL_SA algorithm. Under the multi-objective optimization condition, by setting different parameters, the optimization effect of power and delay can be adjusted to better meet the needs of different applications and improve

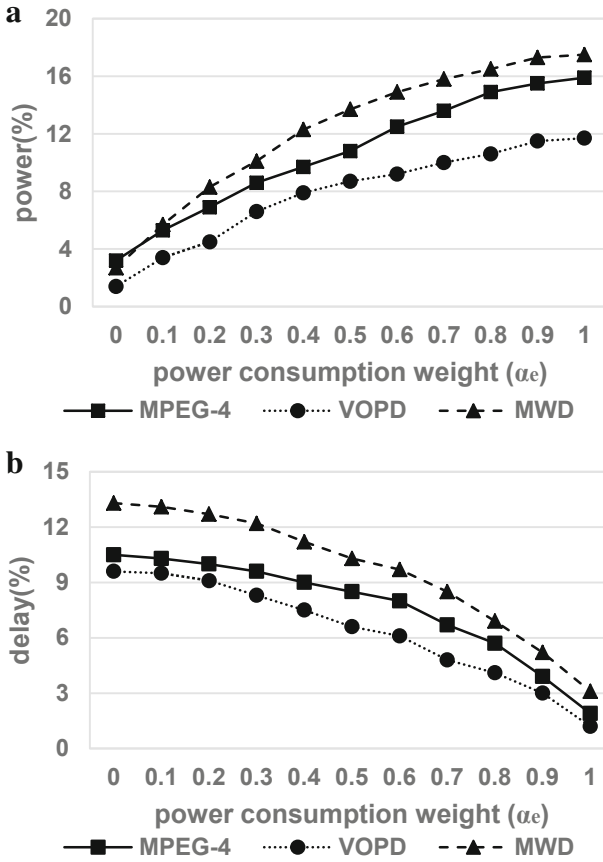


Fig. 5. (a) The power consumption optimization effect. (b) The delay optimization effect.

the flexibility of the mapping scheme. In terms of time complexity, this scheme is determined by the number of application task nodes n , the simulated annealing initial temperature T , the number of iterations of the simulated annealing algorithm $Iter$, the simulated annealing temperature drop rate $Temp_down$, and the simulated annealing minimum temperature T_min . Through calculation, the time complexity can be obtained as: $O^n * Iter * \log_{Temp_down} \frac{T_min}{T}$. It can be seen that the time complexity of this scheme has not increased compared to the simulated annealing algorithm. But NoC power consumption and delay have been greatly optimized, which can better illustrate the effectiveness of this scheme.

5 Conclusions

This paper first analyzes the NoC mapping problem in the heterogeneous multi-core circumstances. Then the power optimization model, delay optimization model and multi-objective optimization model are established, on which basis, a NoC mapping scheme

based on KL_SA algorithm is proposed. The proposed scheme significantly improves the mapping optimization of heterogeneous multi-core NoC. The simulation experiments are carried out on three general applications. The results show that the mapping scheme proposed in this paper brings significant savings on power consumption and delay compared to the original simulated annealing algorithm.

In the future work, we will deeply study the topology of NoC which design a suitable architecture for the mapping scheme to further optimize the system performance.

Acknowledgement. This work is supported by the National Natural Science Foundation of China (Grant No. 61202076), Beijing Municipal Postdoctoral foundation, Chaoyang District Postdoctoral foundation, along with other government sponsors. The authors would like to thank the reviewers for their efforts and for providing helpful suggestions that have led to several important improvements in our work. We would also like to thank all teachers and students in our laboratory for helpful discussions.

References

1. Dai, H., Qu, H., Zhao, J.: QoS routing algorithm with multi-dimensions for overlay networks. *Chisna Commun.* **10**(10), 167–176 (2013)
2. Houtian, W., Xiangjun, Z.Q.X., et al.: Cross-layer design and ant-colony optimization based routing algorithm for low earth orbit satellite networks. *China Commun.* **10**(10), 37–46 (2013)
3. Juan, F., Zhenyu, L., Sitong, L., et al.: Exploring heterogeneous NoC design space in heterogeneous GPU-CPU architectures. *J. Comput. Sci. Technol.* **30**(1), 74–83 (2015)
4. Ren, X., An, J., Gao, Y., et al.: Fusion strategy with genetic and ant algorithms for low-power NoC mapping. *J. Xian Jiaotong Univ.* **46**(8), 65–70 (2012)
5. Zang Mingxiang, Z., et al.: Improved shuffled frog-leaping algorithm for low-power network-on-chip mapping. *J. Xidian Univ.* **42**(1), 118–123 (2015)
6. Qihua, D., et al.: Modified genetic algorithm based method on low-power mapping in network-on-chip[C]. In: *International Conference on Applied Science & Engineering Innovation* (2015)
7. Cui, H., Zhang, D., Song, G.: A novel mapping algorithm for three-dimensional network on chip based on quantum-behaved particle swarm optimization. *Front. Comput. Sci.* **11**(4), 622–631 (2017)
8. Srinivasan, K., Chatha, K.S., Konjevod, G.: Linear-programming-based techniques for synthesis of network-on-chip architectures. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **14**(4), 0–420 (2006)
9. Leary, G., Srinivasan, K., Mehta, K., Chata, K.S.: Design of network-on-chip architectures with a genetic algorithm-based technique. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **17**(5), 674–687 (2009)
10. Chatha, K.S., Srinivasan, K., Konjevod, G.: *Automated Techniques for Synthesis of Application-Specific Network-on-Chip Architectures*. IEEE Press, Piscataway, NJ 2008
11. Ost, L., Mandelli, M., Almeida, G.M., et al.: Power-aware dynamic mapping heuristics for NoC-based MPSoCs using a unified model-based approach. *ACM Trans. Embed. Comput. Syst.* **12**(3), 1–22 (2013)
12. Nadia, N., Carvalho, M.V., da Silva, L., Mourelle, D.M.: Preference-based multi-objective evolutionary algorithms for power-aware application mapping on NoC platforms. *Expert Syst. Appl.* **39**(3), 2771–2782 (2012)

13. Alagarsamy, A., Gopalakrishnan, L.: SAT: a new application mapping method for power optimization in 2D — NoC. In: International Symposium on Vlsi Design and Test, pp. 1–6 (2016)
14. Lang, H., Liu, A., Xie, M., Wang, T.: UAVs joint vehicles as data mules for fast codes dissemination for edge networking in smart city. *Peer-to-Peer Networking Appl.* **12**(6), 1550–1574 (2019)
15. Wang, T., Wang, P., Cai, S., Ma, Y., Liu, A., Xie, M.: A unified trustworthy environment establishment based on edge computing in industrial IoT. *IEEE Trans. Ind. Inf.* **16**(9), 6083–6091 (2019)
16. Jiang, S., Zhou, J., Lu, Z., et al.: A fault-aware low-power-dissipation dynamic mapping algorithm based on NoC1. In: 2017 20th International Conference on Electrical Machines and Systems (ICEMS), pp. 1–5 (2017)
17. Taassori, M., Niroomand, S., Uysal, S., Vizvari, B., Hadi-Vencheh, A.: Optimization approaches for core mapping on networks on chip. *IETE J. Res.* **64**(3), 394–405 (2018)
18. Qianqi, L., et al.: A multiobjective scatter search algorithm for fault-tolerant NoC mapping optimisation. *Int. J. Electron.* **101**(8), 1056–1073 (2014)
19. Rabindra, K., Srivastava, P., Sharma, G.K.: Network-on-chip: on-chip communication solution. *Int. Rev. Comput. Softw.* **5**(1), 22–33 (2010)
20. Niu, G.: The optimization of the search scheme by the simulated annealing algorithm. In: 2016 6th International Conference on Machinery, Materials, Environment, Biotechnology and Computer (2016)
21. Antunes, C.H., Henriques, C.O.: Multi-objective optimization and multi-criteria analysis models and methods for problems in the energy sector. In: Greco, S., Ehrgott, M., Figueira, J.R. (eds.) *Multiple Criteria Decision Analysis*. ISORMS, vol. 233, pp. 1067–1165. Springer, New York (2016). https://doi.org/10.1007/978-1-4939-3094-4_25
22. Medhi, D., Ramasamy, K.: Network routing, pp. 30–63 (2018)
23. Jiang, N., Becker, D.U., Michelogiannakis, G., et al.: A detailed and flexible cycle-accurate network-on-chip simulator. In: 2013 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS), IEEE (2013)